Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Computers in Biology and Medicine 140 (2022) 105112

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

A Covid-19's integrated herd immunity (CIHI) based on classifying people vulnerability

Asmaa H. Rabie [a,*], Ahmed I. Saleh [a], Nehal A. Mansour [b]

[a] Computers and Control Dept. Faculty of Engineering Mansoura University, Mansoura, Egypt
[b] Nile Higher Institute for Engineering and Technology, Artificial Intelligence Lab., Mansoura, Egypt

ARTICLE INFO

Keywords:
Covid-19
Classification
KNN
Feature selection

ABSTRACT

Unfortunately, Covid-19 has infected millions of people very quickly, and it continues to infect people and spreads rapidly. Although there are some common symptoms of Covid-19, its effect varies from one individual to another. Estimating the severity of the infection has become a critical need as it can guide the decision makers to take an accurate and timely response. It will be valuable to provide early warning before infection takes place about susceptibility to the disease, especially since the lack of symptoms is a feature of the Covid-19 pandemic. Asymptomatic patients are considered as "silent diffusers" of the virus; hence, detecting people who will be asymptomatic before actual infection takes place will certainly safe the society from the uncontrolled and unseen spread of the virus. People can be classified based on their vulnerability to Covid-19 even before they are infected. Accordingly, precautionary measures can be taken individually based on the persons' Covid-19 susceptibility. This paper introduces a Covid-19's Integrated Herd Immunity (CIHI) strategy. The aim of CIHI is to keep the society safe with the minimal losses even with the existence of Covid-19. This can be accomplished by two basic factors; the first is an accurate prediction of the cases who will be asymptomatic if they were infected by the virus, while the second is to take suitable precautions for those who are predicted to be badly affected by the virus even before the actual infection takes place. CIHI is realized through a new classification strategy called Distance Based Classification Strategy (DBCS) which classifies people based on their vulnerability to Covid-19 infection. The proposed DBCS classifies individuals into six different types, then suitable precautionary measures can be taken for every type. DBCS can also identify future symptomatic and asymptomatic cases. In fact, DBCS consists of three sequential phases, which are; (i) Outlier Rejection Phase (ORP) using Hybrid Outlier Rejection (HOR) method, (ii) Feature Selection Phase (FSP) using Hybrid Feature Selection (HFS) method, and (iii) Classification Phase (CP) using Accumulative K-Nearest Neighbors (AKNN). DBCS has been compared with recent Covid-19 diagnosing techniques based on "NileDS" dataset. Experimental results have proven the efficiency and applicability of the proposed strategy as it provides the best classification accuracy.

1. Introduction

Covid-19, this strange virus and its rapidly spreading deadly mutations have affected the daily life of the entire world population [1–3]. Moreover, its impact is expected to continue in the future. Before a vaccine is fully available, scientist's best hope of combating Covid-19 lies in preventing its spread. Unfortunately, till now, the nature and behaviors of Covid-19 are not completely clear. The clinical and epidemiological characteristics of the virus must continue to be investigated as the virus is highly transmissible through humans [1,2,4]. Generally, disease diagnosis depends on clinical symptoms and signs [5]. However, there is definite evidence that many patients infected with Covid-19 are asymptomatic or have too few symptoms to be recognized. The known prevalence of asymptomatic patients is about 19.2% [6,7]. Unfortunately, asymptomatic transmission acts as a silent harbor for the virus infection [6]. As there are difficulties in screening for asymptomatic infection, it is difficult to prevent and control this epidemic at the national level unless it is dealt with appropriately. Before that, it is necessary to know the detailed picture and characteristics of the asymptomatic Covid-19 patients.

Detecting asymptomatic patients is a task that may seem impossible to achieve because this needs continuous scanning of all people to

* Corresponding author. Mansoura University Faculty of Engineering, Egypt.
E-mail address: asmaa91hamdy@yahoo.com (A.H. Rabie).

https://doi.org/10.1016/j.compbiomed.2021.105112
Received 18 October 2021; Received in revised form 2 December 2021; Accepted 2 December 2021
Available online 7 December 2021

discover the presence of the virus. Therefore, there is an urgent need for new mechanisms to discover or anticipate who may be infected with the virus without showing symptoms, as their bodies contain mechanisms that may not be known to resist the virus. Hence, the virus does not affect them, but they remain infectious to others throughout the recovery period during which no symptoms appear. Herd immunity takes place when the virus cannot spread as it continues to encounter people who are protected from infection. There is no need for every member of the population to be immune, just enough people need to be immune. Only when a sufficient part of the population is not susceptible to infection, the potential for an epidemic to spread is minimal. Herd immunity works to control and avoid the spread of a disease in a population only when a specific amount of that population, called Immunity Threshold (IT), becomes immune to the disease either through vaccination or infection and recovery. Hence, when IT is reached, susceptible individuals become protected from infection because the ongoing spread of disease is limited. However, in the case of Covid-19, applying traditional herd immunity gives a false promise to minimize the spread of infection. Accordingly, much attention should be given to introduce a new herd immunity model that has the ability to; (i) accurately predict the asymptomatic Covid-19 cases to reduce the virus transmission as their potential to spread the virus cannot be ignored, (ii) reduce the virus clustered transmission for community and family, and (iii) protect those who are expected to be severely affected by the virus before actually infection occurs, hence, suitable precautionary measures can be taken.

Many machine learning and artificial intelligence methods have been successfully applied to predict and diagnose Covid-19 cases using the clinical and healthcare data [1–4]. However, early prediction frameworks for the impact of Covid-19 on people before actual infection occurs have not yet been implemented. Such frameworks can be helpful in taking proactive measures to combat the spread of the virus as well as to protect individuals from any predicted harmful impact of the virus. They can also be used to implement a new herd immunity model for Covid-19. Data mining (DM) is the process of extracting useful information from large amount of databases [8–13]. DM techniques have been successfully implemented in healthcare domain [4]. Medicinal data mining can utilize the veiled patterns presented in huge medical data. Several DM techniques are useful to medical data such as; association rule mining for finding frequent patterns, classification, and clustering. These techniques are useful in predicting heart diseases, breast cancer, lung cancer, diabetes, and recently Covid-19 [1–4].

The main contribution of this paper is to introduce a suggested plan that will allow society to stay normal as possible even with the existence of Covid-19, while taking some procedures to protect those who are most at risk. This can be accomplished by replacing the traditional herd immunity with a new specialized immunity model for Covid-19, which is called CIHI. The proposed CIHI would essentially allow the coronavirus to run its course with minimal losses. It has the ability to detect people who will be asymptomatic if they were infected by Covid-19. These people are categorized as less dangerous because their bodies can resist the virus, hence, they can be allowed to return to their normal life with continuous follow. On the other hand, people at high risk could be protected through strict measures even if they have not been infected yet. The proposed CIHI is realized using a new classification strategy called DBCS, which aims to classify people into six classes based on their vulnerability by Covid-19. Hence, special procedures and precautionary measures are applied for each type. DBCS has the ability to discover the people expected to be asymptomatic if they were infected with the virus. It can also identify those people who will be badly influenced by the virus.

DBCS consists of three sequential phases, which are; ORP, FSP, and CP. In this work, there are three main contributions represented by the proposed methods presented in ORP, FSP, and CP:

- During ORP, the task is to reject those data items with features that are very different from expectation, which are called outliers. Thus, a new technique called HOR has been introduced for rejecting outliers based on a hybrid method that combines standard division as a statistical method and Improved Binary Particle Swarm Optimization (IBPSO) as a machine learning method.
- On the other hand, the main objective of FSP is to select the best set of features that allows to build a useful classification model. HFS method is applied in FSP to select the most appropriate set of features based on a hybrid method that includes Chi-square as a filter method and Improved Binary Gray Wolf Optimization (IBGWO) as a wrapper method.
- Finally, the proposed AKNN method has been used in CP as a classification model. During CP, a new test case can be classified into one of six classes based on the expected resistance of his body against the Corona virus if he is infected with it, which is an indication of the extent of the damage that the virus may cause if the person is actually infected.

This definitely allows specific precautionary measures to be taken for each class before the occurrence of infection, which can achieve the principle of herd immunity. Hence, some people are allowed to be presented in human gatherings with constant observation and follow-up, others are prevented from the existence in crowded groups, while some people are forced to being at home. DBCS has been compared with modern techniques used to diagnose Covid-19 patients. Experimental results have proven the efficiency and applicability of the proposed strategy because it introduces the best evaluation values in term of accuracy, error, precision, recall, and run time. And accordingly, this guarantees the availability of achieving heard immunity. The remainder of the paper is organized as follows; section 2 provides the required background and basic concepts. Section 3 presents the problem definition and the suggested solution. Section 4 discusses the previous effort about Covid-19 classification models. Section 5 introduces the proposed integrated herd immunity. Next, section 6 presents the experimental setup and results while section 7 discusses the DBCS pros and cons. Finally, section 8 concludes the study and section 9 outlines the main directions for future work.

2. Background and basic concepts

This section aims to present the basic knowledge and emerging ideas on topics relevant to the subject of the paper. Initially, a brief discussion about the infection spread of Covid-19 is presented. Then, traditional herd immunity principles are introduced.

2.1. Hidden demons behind Covid-19 infection spread

Unfortunately, unusually for disease management, the Covid-19 virus has many exceptions that have yet to be explained. To clarify, a positive test result is not the only criterion for the presence of Covid-19 infection [1–3]. On the other hand, the test is usually a support for a clinical diagnosis, not a substitute. However, there is a lack of clinical supervision, and thus health care professionals know very little about the proportions of people who have positive results and who do not actually show symptoms throughout the course of the infection. In other words, health care professionals know very little about proportions of people who have symptom scarcity (e.g., subclinical) or asymptomatic (e.g., persistence of symptoms later) or after infection (e.g., with fragments of viral Ribonucleic Acid (RNA) that remain detectable from a previous infection). One of the strong reasons for the rapid spread of Covid-19 is that some people infected with it do not develop symptoms and are nonetheless contagious [14,15]. Surprisingly, these people do not appear and do not feel sick, but they transmit the virus without realizing it.

Generally, spread of a disease without illness appearance is called asymptomatic (ASM) transmission. On the other hand, a person with signs of disease is called a symptomatic (SYM) case. However, there is

another term that may cause confusion, which is; presymptomatic (PSYM). Although a PSYM case can be understood as a person who has not yet developed any symptoms, PSYM can also mean ASM. Symptomatic, asymptomatic, and pre-symptomatic stages are illustrated in Fig. 1. As illustrated in such figure, in the pre-symptomatic stage, many people are contagious. Thus, transmission of the virus is still possible even though the disease is not externally manifested. This is why governments require entire families to isolate when one of their members gets sick. Hence, asymptomatic or pre-symptomatic cases have been called "silent diffusers". Hence, a critical task for limiting the fast spread of Covid-19 is the early identification of people who will be asymptomatic cases if they were infected by the virus before the actual infection takes place.

2.2. Covid-19 and herd immunity

Herd immunity (HI), also called "population immunity", is the indirect protection from infectious diseases by making a population immune either through vaccination or through immunity developed from a previous infection. However, attempts to achieve HI by exposing people to the virus are not only scientifically problematic but also unethical. The World Health Organization (WHO) supports access to HI through vaccination, not by allowing a disease to spread through the population, as this would lead to unnecessary infections, suffering, and death. Fig. 2 illustrates herd immunity types, which are; no Immunization and with immunization.

However, in the case of Covid-19, the use of traditional HI may give a false promise to minimize the infection spread for the following reasons; (i) achieving HI through community spread of a pathogen is based on the unproven assumption that individuals who survive infection will become immune, (ii) recently, it has been noticed that people get re-infected with Covid-19 after the initial infection, but how frequently these reinfections take place and whether they lead to less serious illnesses remain open questions, (iii) yet, there is no foolproof way to measure the immunity to Covid-19. Laboratories can test whether individuals have Covid-19-specific antibodies, But they still don't know how long any immunity can last, (iv) after recovery, if infected persons become susceptible to infection again, the community may never reach HI through natural transmission. Vaccination is the only ethical path to HI, however, how many people will need to be vaccinated and how often will depend on many factors, including how effective the vaccine is and how long it continues to protect, (v) the several hidden sources of infections named pre-asymptotic and asymptotic individuals, and (vi) viruses mutate all the time. Several new variants of Covid-19 strain have been discovered recently. For illustration, a new strain has swept across the United Kingdom and has been detected in the United States, Canada, and elsewhere which could be the reason behind the sharp rise in cases there. Researchers and scientists say the new strains have much higher transmissibility than the previous type. This means that more people will be taken to hospital, resulting in overcrowding of hospitals. Once hospitals are full, the quality of care for the sickest patients drops, resulting in higher than expected death rates. For the pre-mentioned reasons, Covid-19 needs special treatments to achieve HI.

3. Problem definition and suggested solution

Despite the similarity in the physiological structure, humans differ from each other in the extent to which they respond to diseases. Generally, a person's response to diseases is closely related to his level of immunity, however, there are many other factors that may significantly affect the susceptibility of the human body to a particular disease compared to others. For example, at the beginning of the Corona pandemic, it was believed that the extent of a person's infection with Covid-19 was closely related to his age and his affliction with chronic diseases such as diabetes pressure [1–4]. However, over time, it is noticed that elderly people suffering from these diseases have been recovered with little effect. On the contrary, many healthy people of middle and lower ages were affected by an enormous effect, sometimes even death [1–4]. Unfortunately, there is no test that can distinguish live Covid-19 virus, therefore, no test for infection is currently available. A person who tests positive with any kind of test may or may not be infectious. Moreover, several issues related to Covid-19 are not yet clear such as viral load, viral shedding, infection, infectiousness, and duration of infectiousness. On the other hand, the response of people's bodies and the extent of their susceptibility to disease varies, despite the close convergence of both in age and health status. Some people do not feel they have been infected with Covid-19, and the infection ended without the person feeling any symptoms or even feeling minor and normal symptoms. However, the status of other people, with similar conditions, may be developed into serious complications.

A total of 634 people tested positive among 3063 tests as at February 20, 2020 on board the Diamond Princess Cruise ship, Yokohama, Japan. Of the 634 confirmed cases, a total of 314 and 320 were reported to be symptomatic and asymptomatic, respectively. The proportion of symptomatic and asymptomatic individuals during the period from 13 to 20 February are illustrated in Fig. 3, which indicates that there exists a clear evidence that a substantial fraction of Covid-19 infected individuals are asymptomatic [16–18]. The relatively high proportion of asymptomatic infections could have public health implications. Hence, more attention should be paid for those asymptomatic cases because they could be the hidden source of the spread of the virus infection.

Classifying people into categories according to the extent to which their bodies will be affected by Covid-19, if they are infected with it, may limit the spread of corona disease and even eliminate it for the following reasons; (i) taking appropriate precautionary measures according to the response of the human body has the greatest effect in protecting the individual from actual disease and (ii) the discovery of people who will not show symptoms of the disease or suffer from minor symptoms (e.g., asymptomatic cases) will have a great impact on limiting the spread of the disease. Those people who would not show symptoms act as time

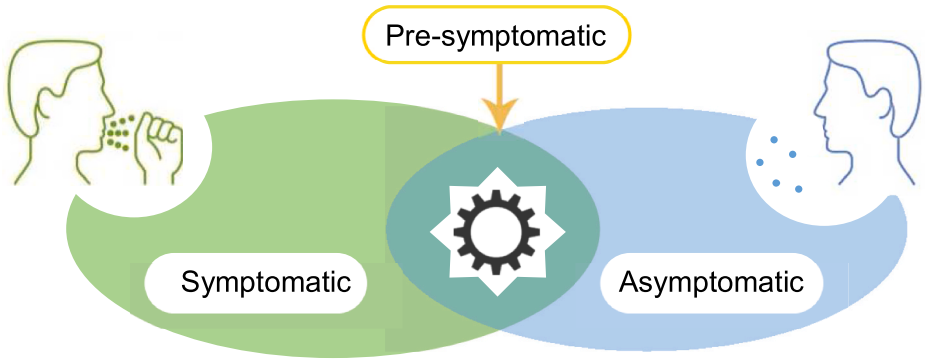


Fig. 1. Symptomatic, Asymptomatic, and Pre-symptomatic transmission.

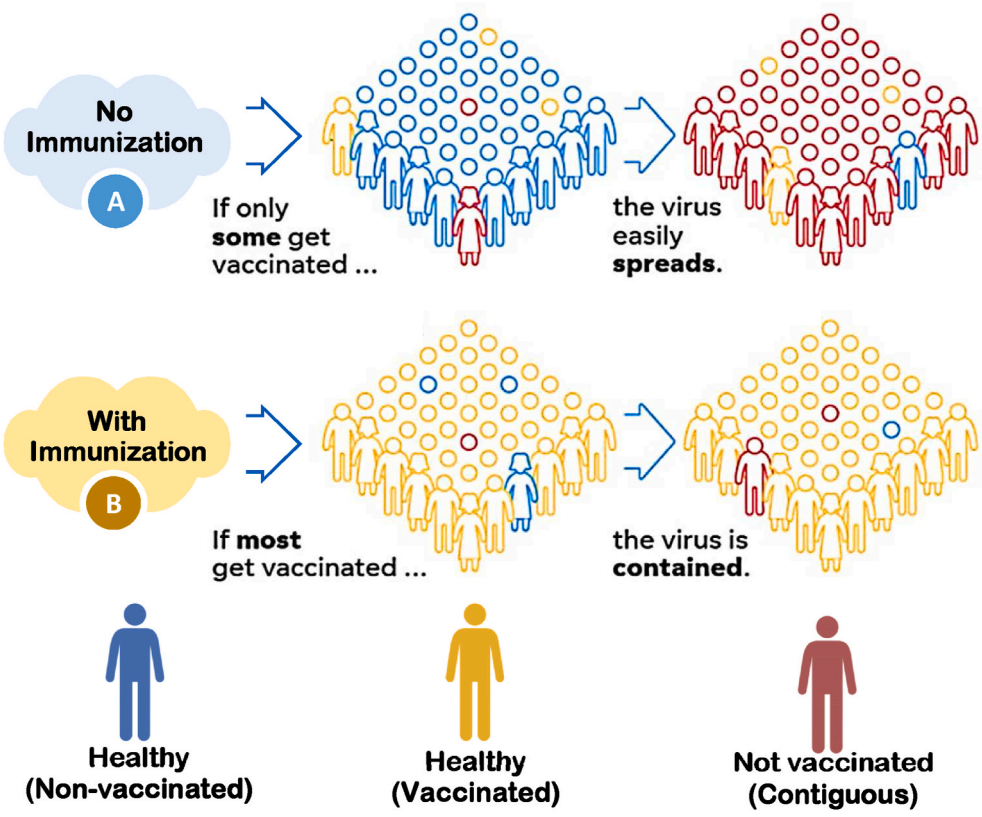


Fig. 2. Herd immunity, (A) No Immunization and (B) With immunization.

Date	Total confirmed cases	Symptomatic cases		Asymptomatic cases	
		No.	%	No.	%
Before 13 Feb.	218	183	83.9%	35	16.1%
On 15 Feb.	285	212	74.4%	73	25.6%
On 16 Feb.	355	244	68.8%	111	31.2%
On 17 Feb.	454	273	60.1%	181	39.9%
On 18 Feb.	542	296	54.6%	246	45.4%
On 19 Feb.	621	307	49.4%	314	50.6%
On 20 Feb.	634	314	49.5%	320	50.5%

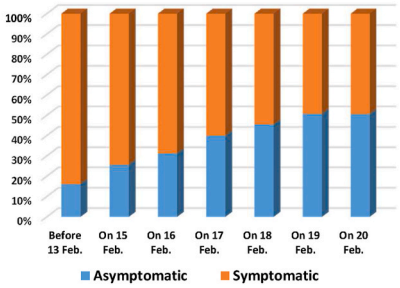


Fig. 3. Symptomatic versus Asymptomatic cases on board the Diamond Princess Cruise ship, Yokohama, Japan, 2020.

bombs as they continue to interact with healthy people and spread the virus without even realizing it. People classification based on their vulnerability level to Covid-19 is presented in Fig. 4. As illustrated in Fig. 4, based on the individual vulnerability level to Covid-19, people can be classified into six types (Type A→F). A person of 'Type A' will not show any symptoms if he infected by the virus, hence, he can be considered as asymptomatic case. 'Type A' will not be affected by the virus but he can spread it, so, he should be identified as he will be a silent spreader of the virus. On the other hand, the remaining types (e.g., Type B→F) are considered as symptomatic but with different vulnerability to the virus. To keep the individuals of the society safe, each type must be subjected to different treatments and rules as illustrated in Table 1.

4. Related work

This section will review the previous research to detect Covid-19 patients. In Ref. [4], a fast and accurate strategy called Distance Biased Nave Bayes (DBNB) was introduced to diagnose Covid-19 contaminated patients. This strategy depended on the results of numerical laboratory tests collected from many healthy and infected people. DBNB introduced two main contributions, which are; a hybrid feature selection method and a new classification method. The hybrid feature selection method includes both filter and wrapper methods to choose the best features for the next classification phase. In the hybrid feature selection method, several filter selection methods have been used to quickly select the best subsets features which are used as initial values for individuals of the particle swarm optimization method used as a wrapper method to accurately select the best features. The new classification method combines both statistical and distance modules called weighted Naïve Bayes (NB) and distance reinforcement modules respectively to address the shortcomings of the classic NB. DBNB provided the best results compared to several modern methods in terms of accuracy, precision, recall, and execution time. Although the DBNB strategy enhanced the performance of the classic or traditional NB, the test-cost measurement should be performed on the proposed particle swarm optimization method to provide the highest accuracy as well as the lowest cost. Additionally, DBNB implementation depended only on using numerical data rather than using nominal data.

As described in Ref. [1], a new strategy known as Feature Correlated Nave Bayes (FCNB) was applied to a dataset containing numerical

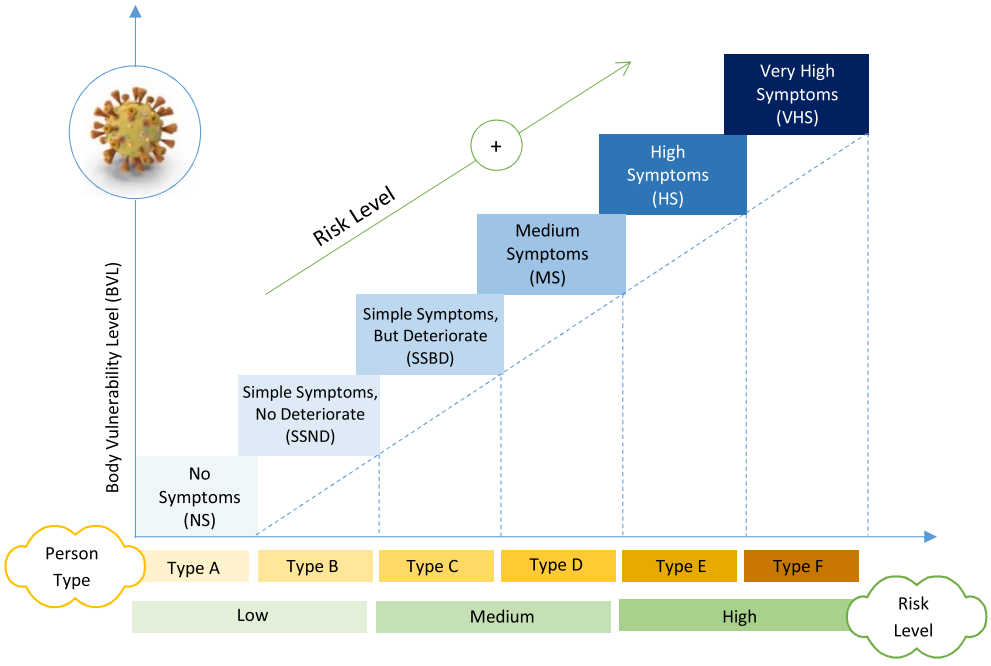


Fig. 4. People classification based on their vulnerability level to Covid-19.

laboratory test results. FCNB is divided into two important stages called pre-processing and classification. Three important phases were used in the pre-processing stage and one phase in the classification stage. The first three phases are as follows; feature selection, feature clustering, and master feature weighting. The feature selection phase was used to choose the most informative features that have an impact on Covid-19. The feature clustering phase was then utilized to aggregate features into groups, with each group termed master feature. Finally, the classification stage depended on a new weighted NB, which had significant advantages. According to the results of the experiments, the FCNB strategy could accurately classify Covid-19 patients. Also, the FCNB could take into account the correlation between features and could minimize the classification time as it takes into account the weights of the used master feature. Although the benefits of FCNB, it did not use outlier rejection method to reject outliers from the dataset before learning the classification method. Additionally, the FCNB suffers from that its implementation depended only on the use of numerical data rather than the use of nominal data.

In [19], a Convolution Neural Network (CNN) model has been used to detect Covid-19 patients using a dataset of Computed Tomography (CT) images. The suggested CNN model had two key algorithms, which are called; CNN architecture and AlexNet, which served as a transfer-learning algorithm. Despite its simplicity, the accuracy of this method is insufficient to accurately diagnose Covid-19 patients. Additionally, it did not use feature selection method or outlier rejection method to filter the data before learning the detection model. Hence, the experiment results demonstrated that utilizing a pre-trained network produced the highest accuracy while using a modified CNN provided the lowest accuracy. In Ref. [2], a new Hybrid Diagnosis Strategy (HDS) was proposed. HDS relied on a new mechanism for rating the chosen features by projecting them into a newly presented patient space. In fact, feature rank was determined by two factors named feature weight and feature's binding degree to its neighbors in the patient space. Then, a hybrid classification model was used to accurately classify the new patient to determine whether or not he was infected. This hybrid classification model used two classifiers called a fuzzy inference engine and a deep neural network. The proposed HDS provided the best recall, precision, accuracy, and F-measure values compared to other recent strategies. The proposed HDS is distinguished by its reliance on using feature selection method to filter the Covid-19 dataset from irrelevant features before learning the classification technique. Thus, it could provide accurate classifications. Additionally, the HDS depended on numerical laboratory tests that have proven effective in detecting Covid-19 patients. Despite the benefits of the proposed HDS, the overall system performance was low because HDS did not take into account the input uncertainty that referred to the effect of driving a simulation with input distributions according to real data.

The Covid-19 Patients Detection strategy (CPDS) has been introduced to provide a more accurate diagnosis of Covid-19 patients as described in Ref. [3]. CPDS introduced two contributions based on CT images of non-Covid-19 persons and Covid-19 patients. The 1st contribution represents a new feature selection approach called Hybrid feature Selection Method (HFSM) which consists of two stages, namely; fast and accurate stages. The goal of the HFSM was to choose the most important features. The 2nd contribution represents an Improved K-Nearest Neighbor (IKNN) classification model, which focuses on evaluating the degree of both closeness and strength of each neighbor of the tested item before selecting just the qualified neighbors for classification. The CPDS could accurately detect infected patients with minimum time penalty. Although the benefits of the proposed CPDS, it did not use outlier rejection technique to remove outliers. Also, CPDS implementation depended on using CT image data rather than using numerical laboratory tests.

In [20], AM-SdenseNet model that includes Convolution Block Attention module and Depthwise Separable Dense Convolutional Network was introduced to diagnose Covid-19 patients based on the CT images dataset. In this work, a publicly available data set COVID-CTx that includes the largest number of positive cases has been established. The proposed AM-SdenseNet outperformed several techniques through experiments where it could provide fast and accurate diagnosis which extremely important to reduce the spreading of infection. Although the COVID-CTx dataset enabled the diagnosis model to provide fast and accurate results, large networks still could not be trained. Additionally, it did not use feature selection or outlier rejection method to correctly learning the diagnosis model. It has not been tested on a different types of datasets such as blood tests. As provided in Ref. [21], the ResNet-50 deep learning model as a new artificial intelligent system was used to diagnose Covid-19 patients based on the 3D CT images. The

Table 1
People classification based on their vulnerability level to Covid-19.

Type	Description	Risk level	Treatment	Case
Type A	No Symptoms (NS)	Low	• To eliminate virus spread, persons of Type A need continuous follow-up and periodic examination, where he/she may be infected with Corona, despite the absence of symptoms. • By making sure of constant observation, a person of type A can be allowed to be in crowded places. • It is preferred to receive the vaccine if it is available.	Asymptomatic
Type B	Simple Symptoms, No Deteriorate (SSND)		• No need for continuous follow-up, but Home isolation is necessary as soon as symptoms appear. • A person of type B can be allowed to be in crowded places. • Simple patient treatments can be followed whenever the symptoms appear. • It is preferred to receive the vaccine if it is available.	Symptomatic
Type C	Simple Symptoms, But Deteriorate (SSBD)	Medium	• The same treatments as SSND, but more serious patient treatments can be followed whenever the symptoms appear.	
Type D	Medium Symptoms (MS)		• Precautionary measures must be applied, such as staying at home. • A person of type D is not allowed to be in crowded places. • Serious patient treatments can be followed whenever the symptoms appear. • For persons of type D, Vaccination is recommended.	
Type E	High Symptoms (HS)	High	• Persons of Type E (or F) must receive the vaccine as soon as possible. • Strict precautionary measures must be applied, he/she must staying at home.	
Type F	Very High Symptoms (VHS)			

evaluation results showed that the ResNet-50 deep learning model outperformed the 3DResNet18 and 3D-ResNet50 models as it could provide accurate results. Although the benefits of the ResNet-50 model, it needed to run several ResNet-50 architectures and also it needed a high memory that might prevent the use of this model in many applications like the mobile telemedicine networks. Finally, modern Covid-19 diagnostic strategies have proven effective in diagnosing Covid-19 patients, but are not being used to classify people based on their vulnerability by Covid-19 yet. In this paper, several recent Covid-19 diagnostic strategies will be compared to our proposed Distance Based Classification Strategy (DBCS) strategy to classify people based on their vulnerability by Covid-19. Table 2 shows the recent related work for Covid-19 diagnosis strategies.

5. The proposed Distance Based Classification Strategy (DBCS)

The proposed Distance Based Classification Strategy (DBCS) will be detailed through this section. DBCS aims to classify people into six classes based on their vulnerability by Covid-19 in which special procedures and precautionary measures are applied for each type. Really, DBCS consists of three sequential phases, which are; ORP, FSP, and CP as shown in Fig. 5. In ORP, HOR method will be provided in order to quickly and accurately reject outliers as possible based on using standard division and IBPSO. FSP aims to select the best features using HFS method that consists of Chi-square as a filter technique and IBGWO as a wrapper technique. These three phases will be depicted in details in the next sub-sections.

5.1. Outlier rejection phase (ORP)

Outlier rejection is the process of finding data items that are very different from expectation in training dataset [8–13]. The existence of outlier items in the input training set may cause unexpected behavior of the used classifier during the testing phase. In DBND, ORP aims to detect outlier items in training dataset and then reject those data to make the diagnostic model able to quickly give accurate values. In fact, outlier items can decrease the accuracy of the classification model. Thus, it is an essential process to eliminate the subset of rare data before starting to train the classification model [8–13]. Many popular rejection methods are applied to evolve the performance of classification model. These rejection methodologies are classified to three main groups, called; cluster-based methodologies, neighbor-based methodologies, and statistical-based methodologies [23–26]. In fact, statistical-based methodologies and distance-based methodologies that belong to neighbor-based methods represent the most popular outlier rejection techniques. Although these popular techniques are simple and fast, they may cannot accurately remove outlier items.

During this section, an effective outlier rejection method called Hybrid Outlier Rejection (HOR) method is provided to reject outlier items. HOR aims to enable the classification model to quickly and accurately perform its tasks. The proposed HOR method mainly composes of two stages, which are called; Fast Rejection (FR) and Accurate Rejection (AR) stages as shown in Fig. 6. In FR stage, standard division is used as a statistical-based methodology to quickly eliminate outliers from the training dataset as possible [22,25]. In AR stage, Improved Binary Particle Swarm Optimization (IBPSO) method is used as an optimization method to accurately reject the rest of outliers in the data to evolve the performance of the classification model [26,27]. Although IBPSO can accurately reject outlier items in the training dataset, it suffers from the computational time. Thus, standard division method in FR stage has been preceded IBPSO to quickly eliminate outlier items before passing the medical dataset to IBPSO method in AR stage. This process aims to reduce the execution time of IBPSO and to provide a robust training dataset without outlier items.

Thus, the proposed HOR aims to quickly and accurately reject outliers from the training dataset before learning the diagnostic model. At the end, the optimal subset of training data will be used to enable the classification model to perform its tasks well. Although IBPSO has been widely applied in many works, it is used in this work as an outlier rejection method by using a credible fitness function called Small Average Distance (SAD) that represents a distance-based outlier rejection method. Accordingly, AR stage implements an optimization technique called IBPSO based on SAD methodology as a credible fitness function. Hence, AR stage contains a hybrid method that includes IBPSO as an optimization technique [26,27] and SAD as an outlier rejection technique [28]. Finally, HOR represents a hybrid rejection method that mainly consists of two techniques, called; (i) standard division as a statistical outlier rejection method and (ii) IBPSO as an optimization technique that depends on distance-based methodology that is called SAD as a fitness function.

Initially, Particle Swarm Optimization (PSO) was built to address many optimization problems represented in continuous numbers search space. However, several problems such as outlier rejection process cannot occur in continuous search space but it can occur in binary search spaces (discrete form) [26,27]. Hence, Binary PSO (BPSO) is a modified version of PSO to provide solutions to binary problems. Really, BPSO

Table 2
The recent related work for Covid-19 diagnosis strategies.

Technique	Description	Advantages	Disadvantages
Distance Biased Naïve Bayes (DBNB) [4]	**DBNB contains two important methods, called; features selection and diagnosis method. Hence, DBNB begins to select the best features in the collected dataset and then use the diagnosis method to diagnose Covid-19 patients.**	**This technique enhances the performance of traditional Naïve Bayes.**	- It is needed to provide the highest accuracy as well as the lowest cost by performing test-cost on the proposed feature selection method. - DBNB implementation depended only on using numerical data rather than using nominal data.
Feature Correlated Naïve Bayes (FCNB) [1]	**FCNB was provided as a new Covid-19 detection strategy in which it consists of two main stages called pre-processing stage and classification stage. While pre-processing stage aims to select the informative features and then convert them to weight space, classification stage aims to provide accurate diagnoses based on weighted NB with several improvements.**	- The accuracy of Covid-19 patients detection is achieved. - The proposed weighted NB and distance reinforcement modules could overcome the issues of traditional weighted NB. - It taken in the consideration the correlation between features. - It minimizes the classification time as it take into consideration the weights of the used master feature.	- No outlier rejection technique have been used to reject outliers. - FCNB implementation based only on using numerical data rather than using nominal data.
Convolution Neural Network (CNN(model [19]	**CNN was developed to detect Covid-19 patients using a dataset of CT images. The suggested CNN model includes two major algorithms, called; CNN architecture and AlexNet as a transfer learning method.**	**Simple to implement.**	**The proposed model's accuracy is insufficient for diagnosing Covid-19.**
Hybrid Diagnosis Strategy (HDS) [2]	**HDS depended on a new methodology for ranking the elected features by projecting them into an introduced patient space.**	**HDS depended on numerical laboratory tests that have proven effective in detecting Covid-19 patients.**	**Performance of overall system is low because HDS does not take in the consideration the input uncertainty that refers to the effect of driving a simulation with input distributions according to real data.**
Covid-19 Patients Detection Strategy (CPDS) [3]	**CPDS was introduced to detect Covid-19 patients using enhanced KNN classifier based on the most important features. These features were elected using hybrid feature selection technique.**	**CPDS could accurately detect infected patients with minimum time penalty.**	- No outlier rejection technique have been used to reject outliers. - CPDS implementation depended on using CT image data rather than using numerical laboratory tests.
Depthwise Separable Dense Convolutional Network with Convolution Block Attention (AM-SdenseNet) module [20]	**AM-SdenseNet model was introduced to diagnose Covid-19 patients based on the CT images dataset.**	**AM-SdenseNet could provide fast and accurate diagnosis which extremely important to reduce the spreading of infection.**	- Large networks still could not be trained. - it did not use feature selection or outlier rejection method to correctly learning the diagnosis model. - It has not been tested on a different types of datasets such as blood tests.
ResNet-50 deep learning model (ResNet-50) [21]	**The ResNet-50 model as a new artificial intelligent system was used to diagnose Covid-19 patients based on the 3D CT images.**	**The ResNet-50 model outperformed the 3DResNet18 and 3D-ResNet50 models.**	- The ResNet-50needed to run several ResNet-50 architectures - It needed a high memory.

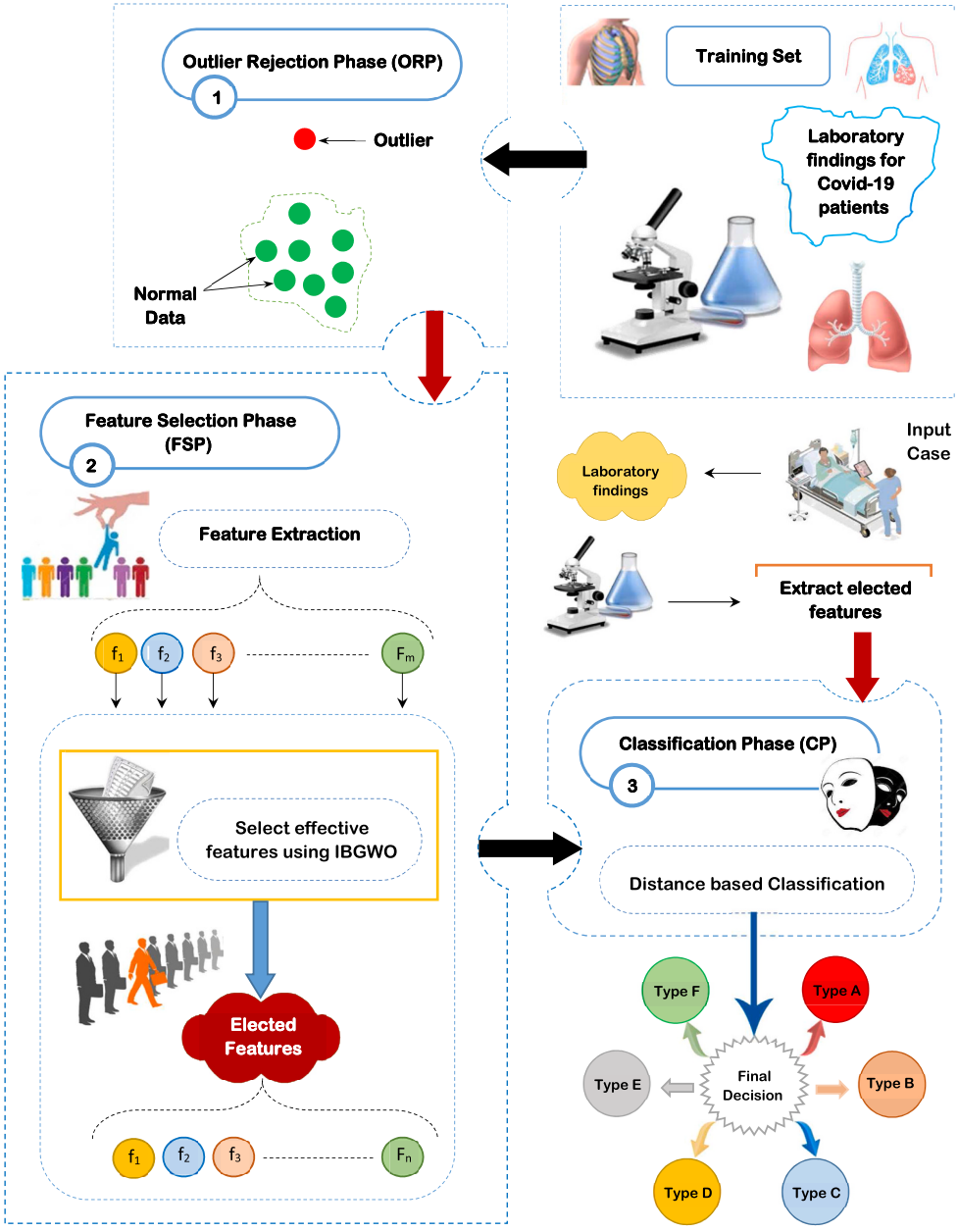


Fig. 5. The proposed DBCS classification strategy.

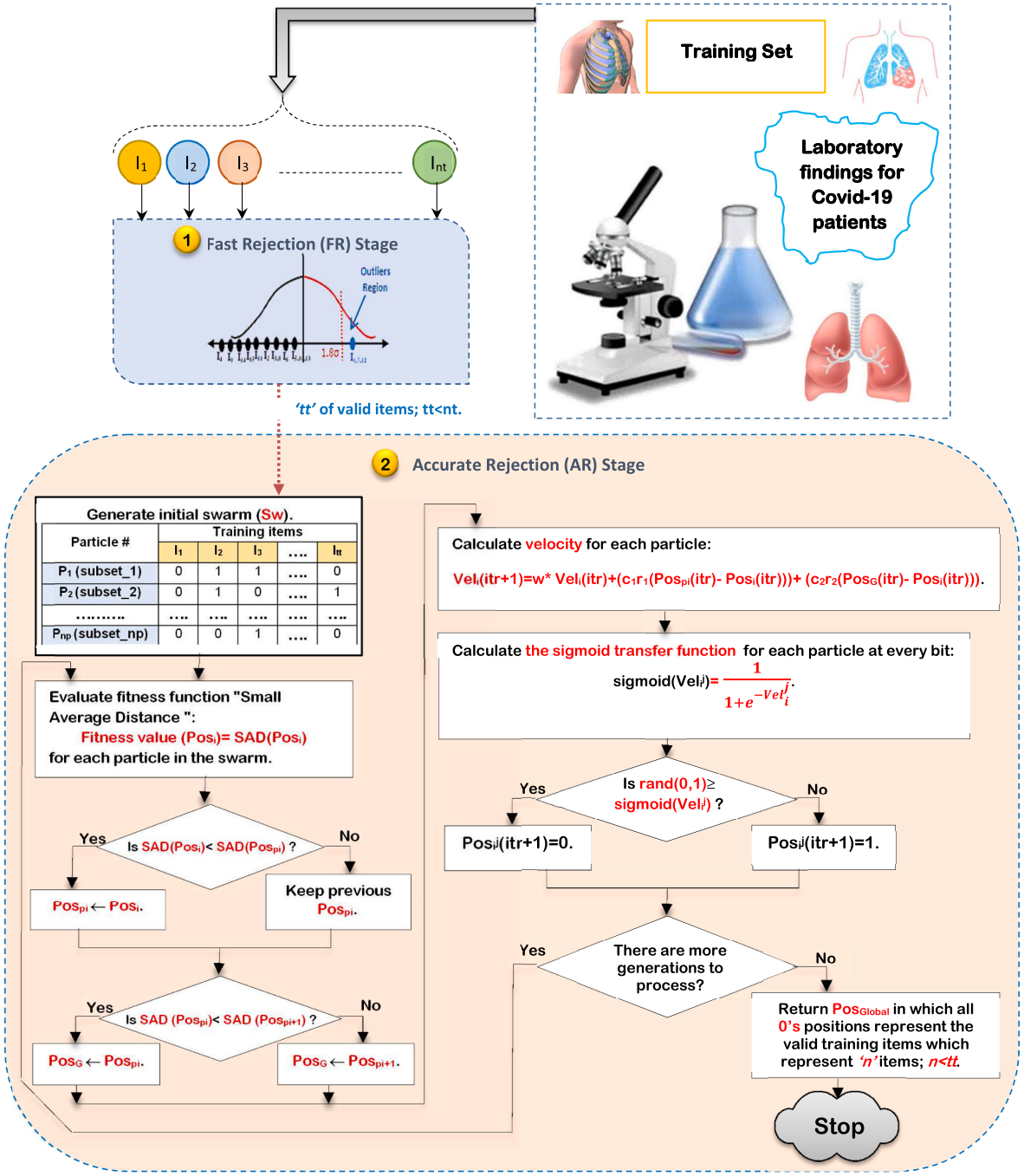


Fig. 6. The sequential steps of HOR method.

Table 3
A representation of each particle.

It_1	It_2	It_3	It_4	It_5	It_6	It_7	It_8	It_9	It_{10}	It_{11}	It_{12}	It_{13}	It_{14}	It_{15}
1	1	1	0	0	1	1	0	1	1	0	1	1	0	1

depends on the use of the sigmoid function that is used to convert the positions of particles in the population into discrete space. Thus, all particles in the population can only have binary values (0 or 1) to cope with the outlier rejection problem that uses zero value as a valid item and one value as an outlier item to make as great as possible the model's performance. While BPSO has the advantages of being adaptable, simple, and flexible which enable it to accurately detect and then reject outlier items in the binary space, it suffers from the computational time.

Consequently, HOR is provided to quickly and accurately detect and then reject outlier items by utilizing the benefits of both standard division as a statistical-based method and IBPSO as an optimization method based on distance-based methodology as a fitness function and tackling their problems. The sequential steps of HOR method using '*nt*' training items are illustrated in Fig. 6. To implement HOR method, the medical dataset should be collected from hospitals. Then, the collected data should be passed to FR stage to implement standard division method to quickly reject outlier items from training dataset as possible (e.g., tt "the number of valid items in the training dataset"), where, $tt < nt$ [22,25]. Then, the training dataset with '*tt*' valid items, which are passed from FR stage, are forwarded to AR stage to enable IBPSO to quickly give accurate subset of valid training items without outliers. Secondly, iterations of IBPSO will continue until discontinuation criteria are met. Finally, the most significant subset of valid training items is presented in the best position called global position in the swarm. SAD, as a fitness function, is applied to evaluate particles in the swarm by calculating the average distance from each particle in the swarm and the center of every class category.

Initially, IBPSO starts with a Swarm (*Sw*) that consists of many particles (birds) as solutions. In IBPSO, each bird or particle in *Sw* represents a potential solution (i.e. a subset of the valid training items in training dataset) in an tt-dimensional search space. Accordingly, a binary string representation is used to represent a subset of valid items in each bird. Each particle's size or length equals the same number of training items in the training dataset. In fact, the bird bits (positions) may contain either zero or one value. The elimination of the *g*th item in the particular subset in the particle as an outlier item can be denoted by one, and the existence of the *g*th item as a valid item can be denoted by zero. An example to clarify the idea, a representation of each particle (bird) is provided in Table 3, suppose $tt = 15$, hence; $It = \{It_1, It_2, It_3, \ldots, It_{15}\}$.

The representation of each bird in tt-dimension (tt = no. of training items in the dataset) is introduced as a vector, $(Pos_i, Pos_{Personal}, Vel_i)$ where Pos_i is the position of *i*th bird; $Pos_i=(Pos_i^1, Pos_i^2, \ldots, Pos_i^{tt})$ and $Pos_{Personal}$ represents the fittest position of the *i*th bird in its history that achieves the best evaluation value; $Pos_{Personal} = Pos_{pi}=(Pos^1_{pi}, Pos^2_{pi}, \ldots, Pos^{tt}_{pi})$. Additionally, Vel_i represents the velocity of *i*th bird; $Vel_i=(Vel_i^1, Vel_i^2, \ldots, Vel_i^{tt})$ and the global position (Pos_{Global}) that indicates to the best position among all the birds in *Sw* is represented as; $Pos_{Global} = Pos_G=(Pos^1_G, Pos^2_G, \ldots, Pos^{tt}_G)$. At the end, Pos_{Global} offers the global optimum solution. Consequently, using IBPSO as an outlier rejection method needs to perform many sequential steps as offered in Fig. 6. In AR stage, the representation of 'n_p' birds in *Sw* is performed and then the evaluation (fitness) function of IBPSO is applied for measuring the evaluation degree of each bird Pos_i (subset of valid training items) based on a distance-based method called SAD. In fact, SAD method is implemented on every particle Pos_i in swarm *Sw* where it represents the aggregated summation of the average distance at every class that is can be calculated using (1).

$$fitness\ value(Pos_i) = SAD(Pos_i) = \sum_{c=1}^{cl} Avg_c \tag{1}$$

where *fitness value* (Pos_i) is the fitness or evaluation value for the position of *i*th particle (Pos). In fact, the fitness function represents a distance based method called SAD. Thus, *fitness value* (Pos_i) can be represented as *SAD* (Pos_i). Avg_C represents the average distance based on the valid training items which belong to class *c* in the *i*th particle; where $c = 1,2, \ldots, cl$. *cl* is the total number of class categories. The best particle provides the lowest fitness value (*SAD*) and vice versa. According to every particle, the average distance Avg_C of the valid items which belong to class *c* can be measured using (2).

$$Avg_C = \frac{1}{q' - z} \sum_{g=1}^{q'-z} Eclid(I_g, Center_c) \tag{2}$$

where *q'-z* are the valid training items in class *c* without *z* items as outliers and $Eclid(I_g, Center_c)$ represents the distance between *g*th item; $I_g = [I_g(f_1)\ I_g(f_2)\ \ldots\ I_g(f_m)]$ and its class center; $Center_c = [Center_c(f_1)\ Center_c(f_2)\ \ldots\ Center_c(f_m)]$ using Euclidean Distance [29]. *g* is an index that refers to the valid training item in the particle which belongs to the class *c*; $g = 1,2, \ldots, q'-z$. Consequently, if item I_g belongs to class *c*, then $Eclid(I_g, Center_c)$ can be calculated using (3).

$$Eclid(I_g, Center_c) = \sqrt{\sum_{j=1}^{m} (I_g(f_j) - Center_c(f_j))^2} \tag{3}$$

where $I_g(f_j)$ is the value of item I_g at the feature f_j. $Center_c(f_j)$ is the center value of class *c* at the feature f_j, and *m* represents the number of features. The center of class *c* according to *j*th feature f_j ($Center_c(f_j)$) is calculated using (4).

$$Center_c(f_j) = \frac{1}{q'} \sum_{g=1}^{q'} I_g(f_j) \tag{4}$$

where $I_g(f_j)$ represents the value of training item I_g that belongs to class *c* at the feature f_j. *q'* represents the number of training items which are belonging to class *c*. IBPSO method searches for the best bird (solution) in order to reduce $SAD(Pos_i)$. Based on evaluation values for the birds in *Sw*, $Pos_{Personal}$ and Pos_{Global} in each particle memory will be updated using (5) and (6) [12,26,27].

$$Pos_{Personal}(Pos_i) = Pos_{pi} = \begin{cases} if(\ SAD(Pos_i) < SAD(Pos_{pi})) \\ Pos_{pi} \quad otherwise \end{cases} \tag{5}$$

$$Pos_{Global} = Pos_G = \begin{cases} Pos_{pi} \quad if(\ SAD(Pos_{pi}) < SAD(Pos_{pi+1})) \\ Pos_{pi+1} \quad otherwise \end{cases} \tag{6}$$

where $Pos_{Personal}(Pos_i)$ is the best solution of *i*th bird that can be denoted as Pos_{pi} that represents the personal fittest position of *i*th bird. Pos_i is the current position of *i*th bird. Additionally, $SAD(Pos_i)$ represents the fitness value (called; small average distance) of the *i*th bird based on its current position. $SAD(Pos_{pi})$ represents the evaluation value of the *i*th bird based on its fittest position. Pos_{Global} represents the fittest bird in whole swarm *Sw* that can be denoted as Pos_G, $SAD(Pos_{pi+1})$ represents the evaluation value of the $(i + 1)^{th}$ bird based on its fittest position, and

Pos_{pi+1} is the personal fittest position of $(i+1)^{th}$ bird. $Pos_{Personal}$ and Pos_{Global} are used for updating every bird's velocity Vel_i in the next iteration $(itr+1)$ using (7) [12,26,27].

$$Vel_i(itr+1) = w * Vel_i(itr) + (c_1 r_1 (Pos_{pi}(itr) - Pos_i(itr))) + (c_2 r_2 (Pos_G(itr) - Pos_i(itr))) \quad (7)$$

Where itr is the current iteration, $Vel_i(itr+1)$ represents the velocity of ith bird at the next iteration, and $Vel_i(itr)$ is the velocity of ith bird at the current iteration. $Pos_{pi}(itr)$ is the personal fittest position of ith bird at the current iteration; $Pos_{Personal}$ (Pos_i) and $Pos_G(itr)$ is the global fittest position in the swarm Sw at the current iteration; Pos_{Global}. Additionally, $Pos_i(itr)$ represents the current position of ith bird at the current iteration. w represents the inertia weight where it is used to control the impact of the previous history of velocities on the current velocity; $w \in$ [0.9–1.2] [12]. c_1 represents the cognitive constant while c_2 is the social acceleration constant; $c_1, c_2 \in$ [2–4]. Both r_1 and r_2 refer to random numbers; $r_1, r_2 \in$[0–1] [12]. Based on the calculation of velocity Vel_i for every bird in Sw, the velocity of bird can refer to the probability distribution with the main role to randomly produce the bird position. Thus, the sigmoid function is implemented to determine a new position of ith bird ($Pos_i(itr+1)$) based on binary values using (8).

$$Pos_i^j(itr+1) = \begin{cases} 0 & if\ rand(0,1) \geq sigmoid(Vel_i^j) \\ 1 & otherwise \end{cases} \quad (8)$$

where $Pos_i^j(itr+1)$ is the value of ith bird at jth position in the next iteration $itr+1$; $j = 1,2,3, \ldots, tt$. Additionally, $rand(0,1)$ represents a random value that belongs to [0,1] and $sigmoid(Vel_i^j)$ indicates to the sigmoid function which represents the probability of jth bit that contains either 0 or 1 value that can be calculated by using (9).

$$sigmoid(Vel_i^j) = \frac{1}{1 + e^{-Vel_i^j}} \quad (9)$$

where the base of the natural logarithm is represented in e. According to the new position $Pos_i(itr+1)$ of every bird in Sw, the evaluation value of every bird is measured using the fitness function in (1). The steps of IBPSO will continue until the finishing condition is met. Finally, the fittest bird of the whole swarm Pos_{Global} represents the solution and the algorithm terminates. All training items denoted by zero in this bird represent the valid items which can be used to accurately learn the classification model (e.g, n "the final number of valid items in the training dataset"), but the training items denoted by one represent outliers which should be removed. Finally, the 'n' of valid items is the best subset of training items used to correctly learn the classification model; $n < tt$. The steps of HOR are presented in Algorithm 1. After eliminating outlier items from training dataset, feature selection process should be implemented to select the most signification features on class category to quickly enable the classification model to give more accurate results. Thus, feature selection process will be applied on dataset without outliers in the next sub-section.

5.2. Feature selection phase (FSP)

In fact, it is not only outlier rejection process is the process that affects the efficiency of the classification model, but also feature selection process has a great effect on improving its efficiency by enabling it to give a faster and more accurate classification [8–13]. The cause of overfitting problem may be the presence of non-informative features in the healthcare dataset [30–32]. Thus, feature selection process will be implemented on dataset during this phase to select the most effective features on the used classifier. Feature selection process can be performed by using filter or wrapper methods [30–32]. While filter methods are simple and fast, these methods may not optimally select the best features. On the other hand, wrapper methods can accurately select subset of informative features that have an impact on the classification method. Recently, optimization techniques are used as feature selection methods to accurately select the best subset of features. Through this section, a Hybrid Feature Selection (HFS) method as a simple but effective selection methodology is introduced to select the best features that enable the classification model to quickly and accurately perform its tasks.

HFS method integrates between wrapper and filter techniques to quickly and accurately eliminate irrelevant features. The proposed HFS method composes of two stages called Fast Selection (FS) stage and Accurate Selection (AS) stage as shown in Fig. 7. In FS stage, Chi-square is used as a filter method to quickly remove non-informative features in the healthcare dataset [10]. In AS stage, Improved Binary Gray Wolf Optimization (IBGWO) method is used as an optimization method to accurately select the best features that can evolve the performance of the classification model [33]. Although IBGWO can accurately eliminate irrelevant features, its execution time is very high. Thus, Chi-square method in FS stage has been preceded IBGWO to quickly eliminate irrelevant features before passing the medical dataset to IBGWO method in AS stage. This process aims to reduce the execution time of IBGWO and to provide a pure dataset without irrelevant features.

Thus, the proposed HFS aims to quickly and accurately select the most effective features in the dataset. Finally, both training and testing dataset based on an optimal subset of features are used to enable the classifier to perform its tasks well. Although IBGWO has been widely applied in many works, it is used in this work as a feature selection method by employing a reliable fitness function that represents the average accuracy value from several classification models (e.g., 'nc' classifiers). The use of many classification models to calculate the fitness degree of each search agent (wolf) aims to generalize the fitness evaluation of each search agent in the population. Accordingly, AS stage uses a machine learning method called IBGWO as a wrapper method based on average accuracy value from 'nc' of classifiers as a reliable fitness function. Finally, HFS is a hybrid method that consists of two important approaches, which are; (i) Chi-square as a filter method and (ii) IBGWO as a wrapper method that depends on average accuracy value from 'nc' of classifiers as a fitness function.

Algorithm 1. Hybrid Outlier Rejection (HOR) Algorithm.

Initially, Gray Wolf Optimization (GWO) built to address many optimization problems represented in continuous numbers search space. However, several problems such as feature selection process cannot

Hybrid Outlier Rejection (HOR) Algorithm

- ***Inputs:***
  - *R= (D,F); training data of 'n_t' items denoted by D={I_1, I_2, I_3, ……, I_{nt}} in which each item $I_j \in D$ is expressed as an ordered set of 'm' features; $I_j(f_1, f_2, f_3, ….., f_m)=[f_{j1}, f_{j2}, f_{j3}, …, f_{jm}]$.*
  - *Input target classes expressed by the set TC={NS, SSND, SSBD, MS, HS,VHS}.*
  - *Six item sets D_{NS}, D_{SSND}, D_{SSBD}, D_{MS}, D_{HS} , and D_{VHS} so that $D=D_{NS}\cup D_{SSND}\cup D_{SSBD}\cup D_{MS}\cup D_{HS}\cup D_{VHS}$. So that D_c expresses the set of items belong to "C" class.*
  - *n_p=No. of particles (birds) in swarm "swarm size".*
  - *$Pos=Pos_1, …., Pos_{np}$; group of particles positions in swarm.*
- ***Output:***
  - *O= the valid training items in the best particle in whole swarm (Pos_{Global}) that provides the minimum fitness value.*
- ***Steps:***

/******* **Implement Fast Rejection (FR) Stage** *******/

1: Detect and then reject *'qq'* of outlier training items using standard division method; $qq=n_t-tt$.

/******* **Implement Accurate Rejection (AR) Stage** *******/

// Construct initial swarm of IBPSO.

2: Randomly generate *'n_p'* of particles in an initial swarm (*Sw*) in tt-dimension with particles positions denoted by (*Pos*); $Pos_i=\{Pos_i^1, Pos_i^2, …, Pos_i^{tt}\}$.

// Calculate fitness degree for each particle's position.

3: **For each $Pos_i \in$ Pos do**

4: $fitness\ value(Pos_i) = SAD(Pos_i) = \sum_{c=1}^{cl} Avg_c$

5: **Next**

// Update the optimum solution of each particle ($Pos_{Personal}$).

6: **For every $Pos_i \in$ Pos do**

7: $$Pos_{Personal}(Pos_i) = Pos_{pi} = \begin{cases} Pos_i & if\ (SAD(Pos_i) < SAD(Pos_{pi})) \\ Pos_{pi} & Else \end{cases}$$

8: **Next**

// Update the optimum particle of the whole swarm (Pos_{Global}).

9: **For every $Pos_i \in$ Pos do**

10: $$Pos_{Global} = Pos_G = \begin{cases} Pos_{pi} & if\ (SAD(Pos_{pi}) < SAD(Pos_{pi+1})) \\ Pos_{pi+1} & Else \end{cases}$$

11: **Next**

// Calculate the new velocity of each particle.

12: **For every $Pos_i \in$ Pos do**

13: $$Vel_i(itr+1) = w * Vel_i(itr) + \left(c_1 r_1 \left(Pos_{pi}(itr) - Pos_i(itr)\right)\right) + \left(c_2 r_2 \left(Pos_G(itr) - Pos_i(itr)\right)\right)$$

14: **Next**

// Calculate the sigmoid function of each particle position.

15: **For every $Pos_i \in$ Pos do**

16: $$sigmoid\left(Vel_i^j\right) = \frac{1}{1+e^{-Vel_i^j}}$$

17: **Next**

Algorithm Parameters	
R	Input training dataset that includes the training items and its features, *R= (D, F)*.
D	The training items; *D={I_1, I_2, I_3, ……, I_{nt}}*.
F	Set of input features in dataset, *F= f_1;……;f_m*.
n_t	No. of training items in the input dataset.
n_p	No. of particles (birds) in swarm "swarm size".
Pos	Group of particles positions in swarm; *$Pos=Pos_1, …., Pos_{np}$*.
O	The valid training items in the best particle that produces global position in whole swarm (Pos_{Global}) that provides the minimum fitness value.
Pos_{Global}	The best position in swarm that provides the minimum fitness value; Pos_G.
Sw	Initial swarm.
qq	No. of rejected outlier items from FR stage.
tt	No. of valid training items produced from FR stage; $tt<n_t$.
c_1,c_2	The cognitive and social acceleration constants; $c_1,c_2 \in [2\text{-}4]$; $c_1+c_2=4$.
r_1,r_2	Uniformly distributed random numbers; $r_1,r_2 \in [0\text{-}1]$.
w	Inertia weight; $w \in [0.9\text{ -}1.2]$.
rand	Uniformly distributed random number; $rand \in [0\text{-}1]$.
Pos_i	The i^{th} particle in the swarm.
$SAD(Pos_i)$	The small average distance of i^{th} particle
$Pos_{Personal}(Pos_i)$	The optimum solution of Pos_i particle; Pos_{Pi}
$Vel_i(itr+1)$	The new velocity of particle Pos_i for iteration itr+1.
$sigmoid(Vel_i^j)$	The sigmoid function of i^{th} particle velocity at j^{th} position.
$Pos_i(itr+1)$	The new position of particle Pos_i for iteration itr+1.

// Calculate the new position of each particle based on binary values.

18: **For every $Pos_i \in$ Pos do**

19: $$Pos_i^j(itr+1) = \begin{cases} 0 & if\ (rand(0,1) > sigmoid(Vel_i^j)) \\ 1 & Else \end{cases}$$

20: **Next**

21: *Updating the values of w, c_1,c_2,r_1, and r_2 parameters according to their corresponding ranges.*

22: **If (there are more generations to process) then**

23: *Go to step3.*

24: **Else**

25: *Return Pos_{Global} in O, where all zeros bits in this particle represents the valid training items.*

26: **End if**

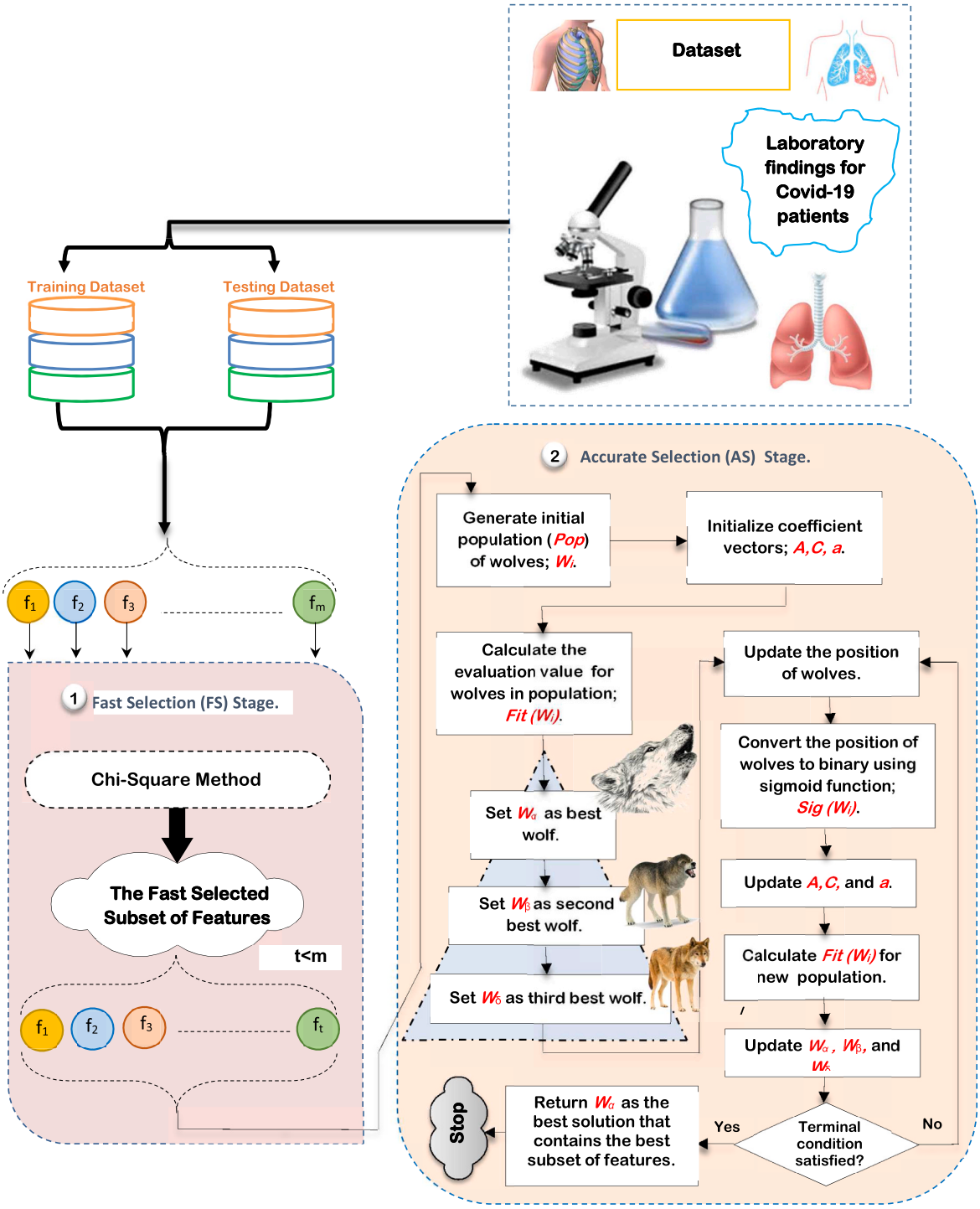


Fig. 7. The sequential steps of HFS method.

Table 4
An example of single search agent.

f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}
0	1	0	1	1	0	1	0	1	1

Table 5
Determine the best search agent based on both every classifier and average accuracy.

Classifier #	Accuracy of every search agent		The best search agent (W_α)
	W_1	W_2	
C_1 =NB	0.75	0.7	W_1
C_2 =KNN	0.9	0.7	W_1
C_3 =SVM	0.8	0.9	W_2
Average accuracy	0.816	0.767	W_1

occur in continuous search space but it can occur in binary search spaces (discrete form) [33–35]. Hence, Binary GWO (BGWO) is a modified version of GWO to provide solutions to binary problems. Really, BGWO depends on using the sigmoid function that is used to convert the positions of the search agents (wolves) in the population from the continuous search space into discrete space. Thus, all search agents in the population can only have binary values (0 or 1) to cope with the feature selection problem that depends on selecting or not selecting the best subset of features to make as great as possible the model's performance. While BGWO has the advantages of being adaptable, simple, and flexible which enable it to accurately select the best subset of features in the binary space, its execution time is very high. Consequently, HFS is provided to quickly and accurately select the best subset of features by utilizing the benefits of both Chi-square as a filter selection technique and IBGWO as a wrapper selection technique and tackling their problems.

To implement IBGWO method, it follows the same steps for implementing the standard BGWO, with the difference that IBGWO is distinguished by using a better fitness function to evaluate every search agent in the population. The fitness function used in IBGWO is the average accuracy value from several classification models trained on the same subset of features in dataset to generalize the evaluation of each search agent in the population. In other words, the calculation of fitness values for search agents in IBGWO depends on several classifiers rather than using only a particular classifier to ensure the generality of the feature selection. Hence, the subset of features which have a significant and effective effect on most classification methods and not for a particular one classifier will be selected to ensure the effectiveness of the selected features on any classification model. Fig. 7 shows the main steps of implementing HFS method using '*m*' features. To implement HFS method, the filtered data passed from the previous phase called ORP should be entered to FS stage for implementing the Chi-square method to quickly select t subset of informative features (e.g., t "the number of selected features in the dataset"), where, t<m. Then, the dataset with 't' features, which are selected from FS stage, are passed to AS stage to enable IBGWO to quickly select the best features as possible. Secondly, iterations of IBGWO will continue until discontinuation criteria are met. Finally, the best search agent in the population called Alpha (α) introduces the most significant features.

Initially, IBGWO starts with a Population (Pop) that consists of many search agents (e.g., "wolves") as solutions. In BGWO, each search agent in Pop is a potential solution (i.e. a subset of the most effective features) in an t-dimensional search space. Accordingly, a binary string representation is used to represent a subset of informative features in each search agent. Each search agent's size or length equals the same number of features presented in the medical dataset. Actually, the search agent bits (positions) may contain either zero or one value. The elimination of the k^{th} feature in the particular subset in the search agent can be denoted by zero, and the selection of the k^{th} feature can be denoted by one. An example to clarify the idea, a single search agent is represented in Table 4, assuming $t = 10$, hence; $F = \{f_1, f_2, f_3, \ldots, f_{10}\}$.

Each search agent in Pop is represented in t-dimension (t = no. of the selected features in FS phase) as a vector that represents the position of i^{th} wolf; $W_i = (W_i^1, W_i^2, \ldots, W_i^t)$. Then, the best three solutions based on their fitness values are assigned to W_α, W_β, and W_δ as Alpha, Beta, and Delta wolves respectively. Through iterations, the coefficient vectors called A and C should be adjusted for the best solutions; W_α, W_β, and W_δ to enable them to update their positions. Based on the updated positions of W_α, W_β, and W_δ wolves (best solutions), the reset of other wolves in Pop can update their positions. Finally, the best solution W_α is used to represent the best subset of features when a termination condition is satisfied. Hence, using IBGWO as a selection method needs to follow several main steps as shown in Fig. 7. In AS stage, 'n_w' search agents (wolves) are represented in Pop and then the evaluation (fitness) function of IBGWO is applied to calculate the evaluation degree of each search agent W_i (subset of input features). The fitness (evaluation) function is the average accuracy value from 'nc' classifiers to ensure that the selected subset of features is the best subset that can enhance the performance of any classifier to provide fast and accurate classification. The fitness function according to 'nc' classifiers can be calculated for i^{th} search agent (W_i) by using (10).

$$Fit(W_i) = \frac{\sum_{j=1}^{nc} Accuracy_j(W_i)}{nc} \tag{10}$$

Where $Fit(W_i)$ is the fitness value for i^{th} search agent, $Accuracy_j(W_i)$ is the accuracy value of j^{th} classifier according to the selected features in i^{th} search agent where j is an index that refers to the used classifiers; j= 1,2,….,nc. *nc* is the number of classifiers used to evaluate the selected features in each search agent. To clarify the idea, assume that there are two search agent in population; $n\_pop = 2$ and three classifiers; $nc = 3$, used to evaluate the selected features in every search agent as presented in Table 5. According to Table 5, it is assumed that the used classifiers

are Naïve Bayes (NB) [12,13], K-Nearest Neighbors (KNN) [3,12], and Support Vector Machine (SVM) [9,13]. Based on their accuracy values for search agents, it is noted that NB and KNN proven that the first search agent (W_1) is better than the second one. On the other hand, SVM proven that the second search agent (W_2) is better than the first one. Finally, the best search agent is the first one based on the average accuracy value. Accordingly, based on a single classifier, to determine the fitness evaluation for search agents cannot generally provide the optimal subset of features that can adaptive with any used classifier. For this reason, the used fitness function in this work based on using the average accuracy value to provide a global solution.

Based on evaluation values for the search agents in *Pop*, the best three solutions; W_α, W_β, and W_δ will be assigned. Then, the other search agents in Pop including Omega (ω) will update their positions based on the position of W_α, W_β, and W_δ. The reason is that W_α, W_β, and W_δ represent the leaders which have better knowledge about the potential position of prey. Hence, ω can be guided by these leaders to move toward the optimal position. Before starting to update the positions of search agents in *Pop*, it is an important to calculate coefficient vectors A and C for the leaders; W_α, W_β, and W_δ by using (11) and (12) [33–35].

$$\vec{A} = |2 * \vec{a} * \vec{r}_1 - \vec{a}| \tag{11}$$

$$\vec{C} = 2 * \vec{r}_2 \tag{12}$$

where $\vec{r}_1$ and $\vec{r}_2$ are random vectors in [0,1] and $\vec{a}$ is the encircling coefficient used to balance the tradeoff between exploration and exploitation. $\vec{a}$ is linearly decreasing from 2 to 0 over iterations by using (13) [33–35].

$$\vec{a} = 2 - 2*(\frac{itr}{Max\_itr}) \tag{13}$$

where itr represents the number of iterations and *Max_itr* represents the maximum number of iterations. After calculating the coefficient vectors A and C for the leaders; W_α, W_β, and W_δ, each search agent (e.g., i^{th} search agent) in Pop can update its position in the next iteration (itr+1) based on $\vec{W}_1$, $\vec{W}_2$, and $\vec{W}_3$ by using (14) [33–35].

$$\vec{W}_i(itr+1) = \frac{\vec{W}_1 + \vec{W}_2 + \vec{W}_3}{3} \tag{14}$$

where $\vec{W}_1$, $\vec{W}_2$, and $\vec{W}_3$ are the positions of leaders α, β, and δ respectively based on the current wolf (W_i). $\vec{W}_1$, $\vec{W}_2$, and $\vec{W}_3$ can be calculated by using (15-17) [33–35].

$$\vec{W}_1 = \vec{W}_\alpha - \vec{A}_1 \vec{D}_\alpha \tag{15}$$

$$\vec{W}_2 = \vec{W}_\beta - \vec{A}_2 \vec{D}_\beta \tag{16}$$

$$\vec{W}_3 = \vec{W}_\delta - \vec{A}_3 \vec{D}_\delta \tag{17}$$

where $\vec{W}_\alpha$, $\vec{W}_\beta$, and $\vec{W}_\delta$ are the position of the leaders (α, β, and δ) at iteration itr and $\vec{A}_1$, $\vec{A}_2$, and $\vec{A}_3$ are the coefficient vector A for α, β, and δ respectively. $\vec{A}_1$, $\vec{A}_2$, and $\vec{A}_3$ are calculated as in (11). Additionally, $\vec{D}_\alpha$, $\vec{D}_\beta$, and $\vec{D}_\delta$ are the distance vectors that calculate how far the leaders (α, β, and δ) are from the i^{th} wolf (W_i). $\vec{D}_\alpha$, $\vec{D}_\beta$, and $\vec{D}_\delta$ can be calculated by using (18-20) [33–35].

$$\vec{D}_\alpha = |\vec{C}_1 . \vec{W}_\alpha - \vec{W}_i| \tag{18}$$

$$\vec{D}_\beta = |\vec{C}_2 . \vec{W}_\beta - \vec{W}_i| \tag{19}$$

$$\vec{D}_\delta = |\vec{C}_3 . \vec{W}_\delta - \vec{W}_i| \tag{20}$$

where $\vec{C}_1$, $\vec{C}_2$, and $\vec{C}_3$ are the coefficient vector C for α, β, and δ respectively which can be calculated as in (12). In fact, the generated position value for each search agent W_i in Pop is a continuous value that cannot be directly used to select the informative features. Thus, the sigmoid function should be used as a transformation function to convert the continuous value to be a binary one. Consequently, every search agent's position; $W_i = (W_i^1, W_i^2, \ldots, W_i^t)$ in Pop should be updated by implementing the sigmoid function to determine new search agent's position based on binary values; $W_{binary\_i} = (W^1_{binary\_i}, W^2_{binary\_i}, \ldots, W^t_{binary\_i})$ using (21) [33].

$$W^k_{binary\_i}(itr+1) = \begin{cases} 1 & if\ rand(0,1) \geq sig(W_i^k) \\ 0 & otherwise \end{cases} \tag{21}$$

where Wk_{binary_i} (itr+1) represents the binary value of i^{th} search agent at *k*th position in the next iteration *itr + 1*; k=1,2,3,…,t. Additionally, rand (0,1) is a random value that belongs to [0,1] and $sig(W_i^k)$ is the sigmoid transfer function that indicates the probability of k^{th} bit that contains either 0 or 1 value calculated by using (22) [33].

$$sig(W_i^k) = \frac{1}{1 + e^{-W_i^k}} \tag{22}$$

where the base of the natural logarithm is represented in e. According to the new position $W^{k}_{binary\_i}$ *(itr+1)* of every search agent in *Pop*, the evaluation value of every search agent is measured using the evaluation function in (10). The steps of IBGWO will continue until the finishing condition is met. Finally, the fittest search agent (W_α) represents the solution and the algorithm terminates. According to W_α as the best solution, all bits denoted by 1 represent the best features that can be used to accurately learn the classification model. After selecting the best subset of features, classification model should be learned based on the filtered data to provide fast and accurate results. The steps of HFS are illustrated in Algorithm 2. After rejecting outliers from training dataset and then selecting the most significant subset of features in the used dataset, classification model should be learned to provide fast and accurate results. Thus, classification process will be applied on dataset without outliers and non-informative features in the next sub-section.

5.3. Classification phase (CP)

The proposed Distance Based Classification Strategy (DBCS) is built upon distance measures. Assuming *cl* target classes $C = \{c_1, c_2, \ldots, c_{cl}\}$. Hence, the distance based probability that an input item *x* belongs to the target class c_i, denoted as; $P_{Dist}(x|c_i)$, relies on measuring the distance among the test item and the training items in an *n* dimensional feature space. Generally, $P_{Dist}(x|c_i)$ is a weighted sum of three different probabilities as illustrated in (23).

$$P_{Dist}(x|c_i) = [\alpha\alpha * P_{DTCC}(x|c_i) + \beta\beta * P_{DTNN}(x|c_i) + \lambda * P_{AKNN}(x|c_i)]/3 \tag{23}$$

Where $P_{Dist}(x|c_i)$ is the distance based probability that the test item *x* belongs to the target class c_i. $\alpha\alpha$,$\beta\beta$, and λ are the considered weights. $P_{DTCC}(x|c_i)$ is the probability based on distance to class center, which relies on the distance between the test item *x* and the center of the target class c_i. $P_{DTNN}(x|c_i)$ is the probability based on distance to nearest neighbors, which relies on measuring the distance from the test item *x* and a specific set of its nearest neighbors. $P_{AKNN}(x|c_i)$ is the probability based on a proposed Accumulative KNN (AKNN) technique. The belonging degree that the input item *x* belongs to class c_i denoted as $Belonging(x|c_i)$ can be calculated by (24).

$$Belonging(x|c_i) = \alpha\alpha * P_{DTCC}(x|c_i) + \beta\beta * P_{DTNN}(x|c_i) + \lambda * P_{AKNN}(x|c_i) \tag{24}$$

Finally, the target class of the input item can be calculated by (25). More details about the used probabilities (e.g., $P_{DTCC}(x|c_i)$, $P_{DTNN}(x|c_i)$,

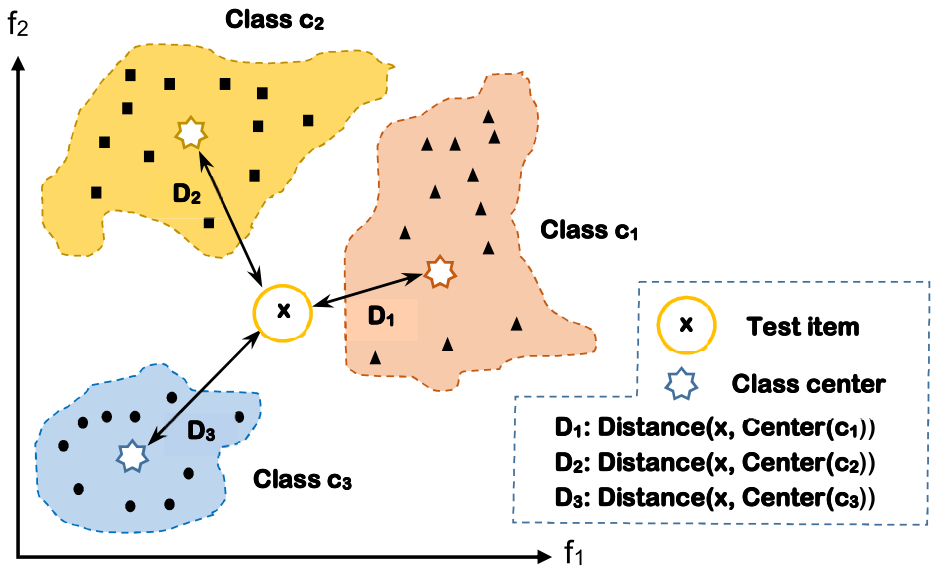


Fig. 8. Calculating Probability based on Distance to Class Center.

and $P_{AKNN}(x|c_i)$ are illustrated in the next sub-sections.

$$Target(x|c_i) = \underset{\forall c_i \in C}{\operatorname{argmax}}(Belonging(x|c_i))$$

$$Target(x|c_i) = \underset{\forall c_i \in C}{\operatorname{argmax}}(\alpha\alpha * P_{DTCC}(x|c_i) + \beta\beta * P_{DTNN}(x|c_i) + \lambda * P_{AKNN}(x|c_i)) \quad (25)$$

where $Target(x|c_i)$ is the target class of the input item x and c_i is an index refers to one of the target classes; $c_i = 1,2, \ldots, cl$.

Algorithm 2. Hybrid Feature Selection (HFS) Algorithm.

5.3.1. Probability based on distance to class center (P_{DTCC})

The distance between the item and the class center can be used as an indication to the degree of correlation between the item and the class. Assuming n dimensional feature space, the Euclidian distance can be calculated using (26). On the other hand, assuming three target classes (e.g., $cl = 3$) and two dimensional feature space (e.g., $n = 2$), Fig. 8 illustrates that the more the distance between the class center and the test item x, the less the correlation between the class and the item. Hence, as Fig. 8 depicts, since $Distance(x,Center(c_i)) < Distance(x,Center(c_j)) < Distance(x,Center(c_k))$, this yields; $Correlation(x,c_i) > Correlation(x,c_j) > Correlation(x,c_k)$. Assuming cl target classes, $C = \{c_1, c_2, \ldots, c_{cl}\}$, then;

$$P_{DTCC}(x|c_i) \propto Correlation(x, c_i)$$

$$Correlation(x, c_i) \propto \frac{1}{Distance(x, Center(c_i))}$$

$$\therefore P_{DTCC}(x|c_i) \propto \frac{1}{Distance(x, Center(c_i))}$$

$$P_{DTCC}(x|c_i) = \frac{\xi}{Distance(x, Center(c_i))}$$

Since

$$\sum_{\forall c_i \varepsilon C} P_{DTCC}(x|c_i) \propto \sum_{\forall c_i \varepsilon C} \frac{1}{Distance(x, Center(c_i))}$$

This yields;

$$\sum_{\forall c_i \varepsilon C} P_{DTCC}(x|c_i) = \xi * \sum_{\forall c_i \varepsilon C} \frac{1}{Distance(x, Center(c_i))}$$

Also since

$$\sum_{\forall c_i \varepsilon C} P_{DTCC}(x|c_i) = 1$$

So,

$$\xi = {}^{1}/_{\left(\sum_{\forall c_i \varepsilon C} \frac{1}{Distance(x, Center(c_i))}\right)}$$

Finally;

$$P_{DTCC}(x|c_i) = \frac{1}{Distance(x, Center(c_i)) * \sum_{\forall c_i \varepsilon C} \frac{1}{Distance(x, Center(c_i))}} \quad (26)$$

where $P_{DTCC}(x|c_i)$ is the probability based on distance to class center, which relies on the distance between the test item x and the center of the target class c_i. $Correlation(x,c_i)$ is a measurement that calculates the correlation between a test item x and the ith class c_i by calculating the distance between the test item x and the center of ith class c_i. $Distance(x, Center(c_i))$ is the distance between the test item x and the center of ith class c_i. ξ is the proportionality constant and c_i is the index that refers to the class number; $c_i = 1,2, \ldots cl$. For illustration, as shown in Fig. 8, assuming three target classes, $C = \{c_1, c_2, c_3\}$, in a two dimensional

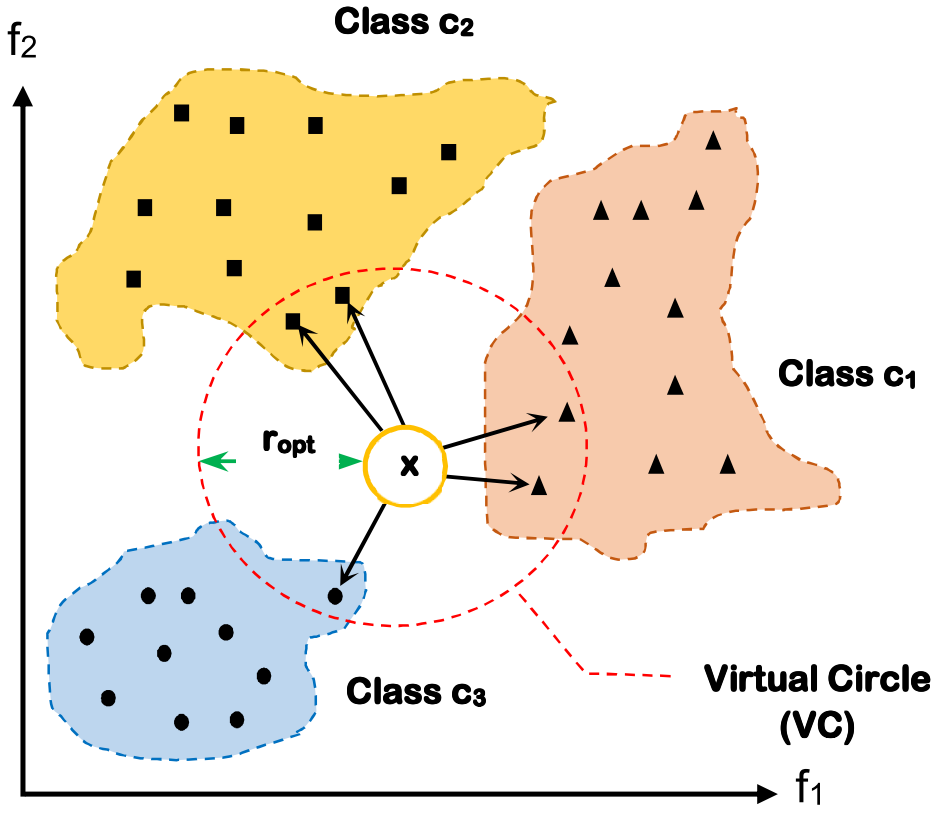


Fig. 9. Calculating probability based on distance to nearest neighbors.

$$r_{opt1} = Distance(x, NCC) \tag{27}$$

$$r_{opt2} = Distance(x, FCC) \tag{28}$$

$$r_{opt3} = \frac{Distance(x, NCC) + Distance(x, FCC)}{2} \tag{29}$$

$$r_{opt4} = \frac{\sum_{i=1}^{n} Distance(x, Center(c_i))}{n} \tag{30}$$

feature space. Based on the data presented in Fig. 8, calculating $P_{DTCC}(x|c_i)\ \forall\ c_i \in C$ using (26) is depicted in Table 6.

5.3.2. *Probability based on distance to nearest neighbors (P_{DTNN})*

Generally, as explained by the traditional K-Nearest Neighbors (KNN) classifier, an input test item can be classified based on the belonging of its neighbors in the n dimensional feature space [3]. However, traditional KNN algorithm suffers from a technical hurdle, which is specifying the optimal value of K. To overcome such hurdle, the test item is expressed side to side with the labeled items in the employed n dimensional space. The input test item is considered as a center of a Virtual Circle (VC) as illustrated in Fig. 9. The optimal radius (r_{opt}) of VC is calculated, then, all labeled items located inside VC are considered for identifying the target class of the input test item. Calculating the optimal radius of VC can be done using four different scenarios as expressed in (27-30).where r_{opt} is the optimal radius of VC which can be done using

Table 6
The procedure used to calculate P_{DTCC} (illustrative Example).

Class	Distance(x,Center(c_j))	$P_{DTCC}(x|c_i)$
C_1	5	$\sum_{\forall c_i \in C} P_{DTCC}(x|c_1) = \frac{1}{5*(\frac{1}{5}+\frac{1}{2}+\frac{1}{3})} = \frac{6}{31}$
C_2	2	$\sum_{\forall c_i \in C} P_{DTCC}(x|c_1) = \frac{1}{2*(\frac{1}{5}+\frac{1}{2}+\frac{1}{3})} = \frac{15}{31}$
C_3	3	$\sum_{\forall c_i \in C} P_{DTCC}(x|c_1) = \frac{1}{3*(\frac{1}{5}+\frac{1}{2}+\frac{1}{3})} = \frac{10}{31}$
$\sum_{\forall c_i \in C} P_{DTCC}(x|c_i)$	1	

four different scenarios; $r_{opt} = \{r_{opt1}, r_{opt2}, r_{opt3}, r_{opt4}\}$. x is the test item, *Distance(x,NCC)* is the distance from the test item to the Nearest Class Center (NCC), *Distance(x,FCC)* is the distance from the test item to the Farthest Class Center (FCC), n is the number of classes. For each target class, a supporters set, denotes as $S(c)$ is formulated, which includes those items that belong to the class c and located inside the proposed VC. For illustration, as depicted in Fig. 10, $S(c_i)$ is the set of items that belong to the class c_i and are also inside VC.

The Belonging Degree (BD) of the test item to a class depends on two basic factors. The first being the number of class supporters, hence, the more the number of class supporters, the more the item belonging degree to the class, while the second element is the average sum of distances between the test item and all the class supporters, hence, the more the average sum of distances between the test item and all class supporters, the less the item belonging degree to the class. This can be formulated through the following equations.

$$BD(x, c_i) \propto |S(c_i)|$$

$$BD(x, c_i) \propto \left(\frac{1}{\left[\frac{\sum_{\forall s_j \varepsilon S(c_i)} Distance(x, s_j)}{|S(c_i)|}\right]}\right)$$

$$BD(x, c_i) = \frac{\mu^*(|S(c_i)|)^2}{\sum_{\forall s_j \varepsilon S(c_i)} Distance(x, s_j)}$$

where μ is the equation constant, assuming $\mu = 1$, then BD is expressed as (31) and P_{DTNN} is formulated as (32).

$$BD(x, c_i) = \frac{(|S(c_i)|)^2}{\sum_{\forall s_j \varepsilon S(c_i)} Distance(x, s_j)} \tag{31}$$

Finally;

$$P_{DTNN}(x, c_i) = \frac{BD(x, c_i)}{\sum_{\forall c_j \varepsilon C} BD(x, c_j)} \tag{32}$$

As depicted in (31) and (32), $S(c_i)$ is the set of items that belong to the class c_i and are also inside VC, $BD(x, c_i)$ is the belonging degree of the item x to class c_i, C is the set of target classes. Fig. 11 gives an illustration for calculating the probability based on distance to nearest neighbors considering 3 target classes c_1, c_2, and c_3. The numbers in the circles indicates the distance from the test item x to the corresponding supporters.

5.3.3. Probability based on accumulative KNN algorithm (P_{AKNN})

In spite of its efficiency and simplicity, traditional KNN suffers from a basic hurdle, which is specifying the optimal value of k. Unfortunately, a minor change in the value of k may totally change the final decision, which is the target class of the input test item. This problem is called KNN trapping, which was illustrated in Ref. [3]. To solve this problem, a new instance of the traditional KNN classifier will be introduced in this section, which is called Accumulative KNN (AKNN). The steps of implementing AKNN are presented in Algorithm 3.

Algorithm 3. Accumulative KNN Algorithm.

The basic idea behind the proposed AKNN is to continuously change the value of k through a pre-defined range and accumulatively calculate a grade for each target class. As illustrated in Algorithm 3, initially, an accumulation limit (final accumulation value), denoted as; ψ is defined, which represents the maximum value of k parameter. The range of ψ can be defined as; $1 < \psi \leq minimum(|c_i|) \; \forall \; c_i \in C$, where $|c_i|$ is the number of items (examples) in the target class c_i, and minimum($|c_i|$) is the number of items (examples) in the class with the minimum number of items (smallest class). Then, an iteration is done to calculate the grade of each target class. Initially, the grade of all target classes is set to zero, hence $\underset{\forall c_i \in C}{Grade}(c_i) = 0$, and the parameter k is set initially to 1. Hence, the iteration starts with $k = 1$ and continues by increasing k by one until k reaches the accumulation limit (e.g., ψ). In each iteration, based on the current value of k, the target class of the input test item is identified, then the grade of the resultant target class is updated (increased by one) accordingly. The iteration is continued until k reaches the accumulation limit (e.g., ψ). At this point the iteration is stopped. The result of the iteration is assigning a grade for each target class. Based on the assigned grade to the target class c_i, the probability, using AKNN, that the input test item x belongs to c_i (denoted as; $P_{AKNN}(x|c_i)$) can be calculated using (33).

$$P_{AKNN}(x|c_i) = \frac{Grade(c_i)}{\psi} \tag{33}$$

where $Grade(c_i)$ is the grade of the target class c_i, and ψ is the accumulation limit. Fig. 12 introduces an illustrative example showing how to calculate $P_{AKNN}(x|c_i) \; \forall \; c_i \in C$ considering three different target classes $C=\{c_1, c_2, c_3\}$. The number of items in c_1, c_2, and c_3 (e.g., $|c_1|$, $|c_2|$, and $|c_3|$) equals 10, 7, and 9 respectively. The accumulation limit (e.g., ψ) equals $\underset{\forall \; c_i \in C}{minmum}|c_i| = 7$. As depicted in Fig. 12, calculating $P_{AKNN}(x|c_i) \; \forall \; c_i \in C$ can be accomplished through three sequential steps. In the first step, the K nearest neighbors to the input test item are identified where $K = \psi$. Then, during the second step, $\underset{\forall c_i \in C}{Grade}(c_i)$ is calculated for each target class starting by $K = 1$, then incrementing K by one until K reaches the accumulation limit (e.g., ψ), which was set previously to 7. After each increment, accumulatively update $\underset{\forall c_i \in C}{Grade}(c_i)$. Finally, based on $\underset{\forall c_i \in C}{Grade}(c_i)$, $\underset{\forall c_i \in C}{P_{AKNN}}(x|c_i)$ can be calculated using (33).

Generally, the basic KNN classifier has two technical hurdles, the first is how to set the optimal value of k, while the second is the KNN laziness. As depicted through the illustrative example shown in Fig. 12, AKNN uses a pre-defined range for the K parameter. This action solves the first technical hurdle for implementing the basic KNN classifier, which is "what is the optimal value of k to be used?". To highlight the basic difference between the traditional KNN algorithm and the proposed AKNN, consider the traditional KNN with k=7. Based on the data illustrated in Fig. 5, the target class of the input test item x will be c_3. However, the proposed AKNN takes another decision to classify x to class c_2 since $P_{AKNN}(x|c_2)$ is the maximum conditional probability. On the other hand, although the proposed AKNN classifier seems to be simple and straight forward, it inherits another technical hurdle of the basic KNN classifier, which is the laziness. The basic KNN classifier is a lazy learner as it consumes long time during the testing phase. However, since the basic KNN uses a static value of k, the calculations are done

Hybrid Feature Selection (HFS) Algorithm

- ***Inputs:***
  - *F= Set of input features in both training and testing dataset; F=f_i;……;f_m*
  - *R= (Dn,F); Training dataset.*
  - *E=(Q,F); Testing dataset.*
  - *m=|F|; No. of feature in the training and testing dataset.*
  - *n_c=No. of search agents (wolves) in population "population size".*
  - *W=W_1….W_{nc}; group of search agents in population.*
  - *n_pop= n_w=No. of search agents (wolves) in population "population size".*

- ***Output:***
  - *O= the selected features in the best search agent called alpha W_α that provides the maximum fitness value.*

- ***Steps:***

/***** Implement Fast Selection (FS) Stage *******/**

1: Select *'t'* of features using Chi-square method, *t<m*.

/***** Implement Accurate Selection (AS) Stage *******/**

// Construct initial population of IBGWO.

2: Randomly generate *'n_pop'* of search agents (wolves) in an initial population (*Pop*) in t-dimension with search agents denoted by (*W*);
W_i={W_i^1, W_i^2,…, W_i^t}.

// Initialize coefficient vectors; a, A,C.

3: Calculate encircling coefficient vector *a* as;

$$\vec{a} = 2 - 2 * \left(\frac{itr}{Max_{itr}}\right)$$

4: Calculate coefficient vectors *A* and *C* for the leaders; W_α, W_β, and W_δ as;

$$\vec{A} = |2 * \vec{a} * \vec{r}_1 - \vec{a}|, \quad \vec{C} = 2 * \vec{r}_2$$

// Calculate fitness degree for each search agent.

5: ***For each search agent $w_i \in W$ do***

6: Calculate the average accuracy value from *'nc'* classifiers as;

$$Fit(W_i) = \frac{\sum_{j=1}^{nc} Accuracy_j\,(W_i)}{nc}$$

7: ***Next***

// Determine the best three solution based on the maximum fitness value.

8: *W_α= the best search agent (the best solution).*

9: *W_β= the second best search agent.*

10: *W_δ= the third best search agent.*

// Update the position of every search agent in Pop based on W_α, W_β, and W_δ

11: ***For each search agent $w_i \in W$ do***

12:

$$\overrightarrow{W_i}(itr+1) = \frac{\overrightarrow{W_1}+\overrightarrow{W_2}+\overrightarrow{W_3}}{3}$$

13: ***Next***

// Calculate the sigmoid function of each search agent position.

14: ***For each search agent $w_i \in W$ do***

15:

$$sig\left(W_i^k\right) = \frac{1}{1+e^{-W_i^k}}$$

16: ***Next***

// Calculate the new position of each search agent based on binary values.

17: ***For each search agent $w_i \in W$ do***

18:

$$W_{binary\_i}^k(itr+1) = \begin{cases} 1 & if \;\; rand(0,1) \geq sig(W_i^k) \\ 0 & otherwise \end{cases}$$

19: ***Next***

20: **If (the termination condition is not satisfied) then**

21: Go to step 3.

22: **Else**

23: Return W_α that provides the maximum fitness value in *O*, where all one's positions in this search agent represents the selected features.

24: **End If**

Algorithm Parameters	
F	Set of input features in both training or testing dataset, F= {f_1;……;f_m }.
R	Training dataset that includes the training items and its features, R= (D, F).
Dn	The training items without outlier items.
E	Testing dataset that includes the testing items and its features, E= (Q, F).
Q	The testing items.
m	No. of features in training and testing data set, m=|F|.
n_pop	No. of search agents in population "population size"; *n_pop=n_w*.
W	Group of search agents in population; *W=W_1….W_{nw}*
O	The selected features in the best search agent called alpha W_α that provide the maximum fitness value.
W_α	Alpha wolf that represents the fittest wolf in the population.
Pop	Initial population.
t	No. of selected features from FS stage; t<m.
a	Encircling coefficient vector.
itr	Index to the current iteration.
Max_{itr}	The maximum iterations number.
A and C	Coefficient vectors for the leaders; W_α, W_β, and W_δ
r_1 and r_2	Random vectors in [0,1].
W_β	Beta wolf that represents the second fittest wolf in the population.
W_δ	Delta wolf that represents the third fittest wolf in the population.
W_1	The position of alpha wolf W_α.
W_2	The position of beta wolf W_β.
W_3	The position of delta wolf W_δ.
W_i(itr+1)	The new position of i^{th} wolf or search agent in the next iteration *itr+1*.
Fit(W_i)	Fitness value of i^{th} search agent W.
$Accuracy_j(W_i)$	The accuracy of j^{th} classifier based on the selected features in i^{th} search agent.
Sig(W_i^k)	The sigmoid function of i^{th} search agent at k^{th} position.
$W^k_{binart\_i}$(itr+1)	The binary position values of i^{th} search agent in the next iteration *itr+1*.

Accumulative K-Nearest Neighbors

- ***Input:***
  - *cl target classes*
    $C=\{c_1, c_2, c_3, \ldots\ldots, c_{cl}\}$
  - *Accumulation limit (final accumulation value), denoted as; ψ, so that,*
    $1<\psi\leq minimum(|c_i|) \quad \forall c_i \in C$
  - *. Input test item to be classified x.*
- ***Output:***
  - $P_{AKNN}(x|c_i) \quad \forall c_i \in C$
- ***Steps***:

1: // Initialize class grades with zero values

2: Grade(c_i)=0 $\forall c_i \in C$

3: // Calculating final classes grades based on accumulation technique

4: ***For k=1 to ψ do***

5: *-Identifying the k nearest neighbors to x*

6: $KNN(x)=\{e_1, e_2, e_3, \ldots\ldots, e_k\}$

7: *- Calculate Distance(x,e_i)* $\forall e_i \in KNN(x)$

8: *- Calculate each class share (CS)*

9: $CS(c_i)=KNN(x) \cap c_i$

10: - Calculate target class of x based on the current value of k

11: **If (multiple classes share the maximum CS) Then**

12: $$\text{Target\_Class}(x,k) = \underset{\forall c_i \in MCSS}{\text{argmax}} \left[\frac{1}{Distance(x,Center(c_i))}\right]$$

13: *Where MCSS is the Maximum Class Share Set, which is a set contains the classes that share the maximum share.*

14: **Else**

15: $$\text{Target\_Class}(x,k) = \underset{\forall c_i \in C}{\text{argmax}}[CS(c_i)]$$

16: **End if**

17: - Update class grades

18: **If (Target_Class(x,k) =c_i) Then**

19: Grade(c_i) ++

20: **End if**

21: ***Next***

22: // Calculating $P_{AKNN}(x|c_i) \; \forall \; c_i \in C$

23: **For each $c_i \in C$ Then**

24: $$P_{AKNN}(x|c_i) = \frac{Grade(c_i)}{\psi}$$

25: **Next**

Algorithm Parameters	
n	Number of target classes
ψ	Final accumulation value
P_{AKNN}	Probability based on accumulative KNN
x	The test item to be classified
C	The set of target classes
c_i	The i^{th} target class in the set C.
$\|c_i\|$	The number of elements (examples) in c_i.
Grade(c_i)	The grade of the class c_i
KNN(x)	The K nearest neighbors to the test item x.
e_i	The i^{th} example in the set KNN(x).
Distance(x,e_i)	The distance between the test item x and the i^{th} example in the set KNN(x).
CS(c_i)	The class share of c_i, which is the set of intersection between KNN(x) and c_i
Target_Class(x,k)	The target class of the test item x considering k nearest neighbors.
MCSS	The Maximum Class Share Set, which is a set contains the classes that share the maximum share.

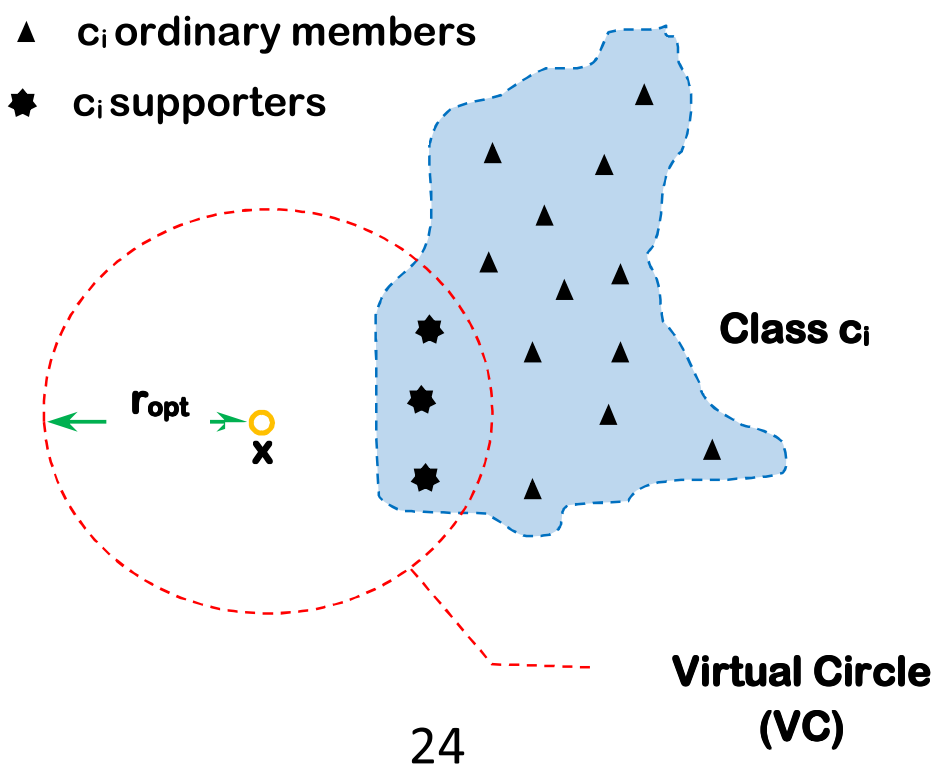
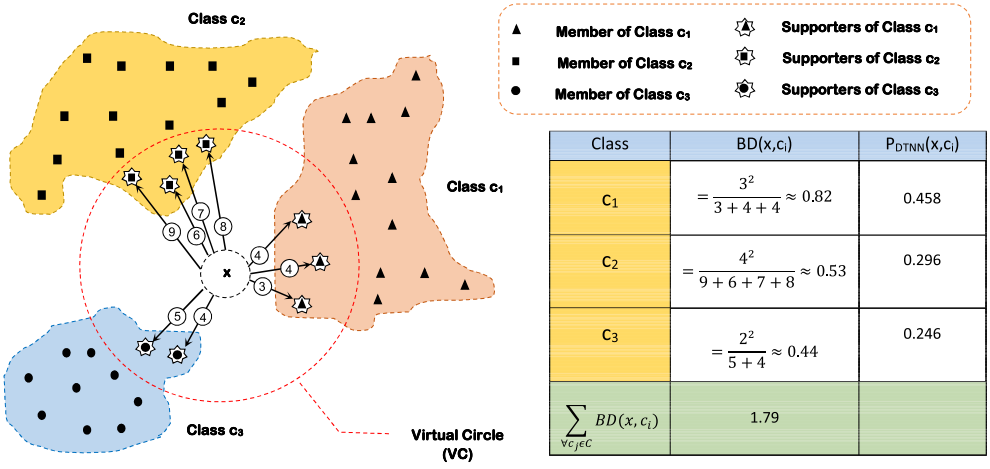


Fig. 10. Class supporters.



Class	BD(x,ci)	PDTNN(x,ci)
C1	$= \frac{3^2}{3+4+4} \approx 0.82$	0.458
C2	$= \frac{4^2}{9+6+7+8} \approx 0.53$	0.296
C3	$= \frac{2^2}{5+4} \approx 0.44$	0.246
$\sum_{\forall c_j \epsilon C} BD(x, c_i)$	1.79	

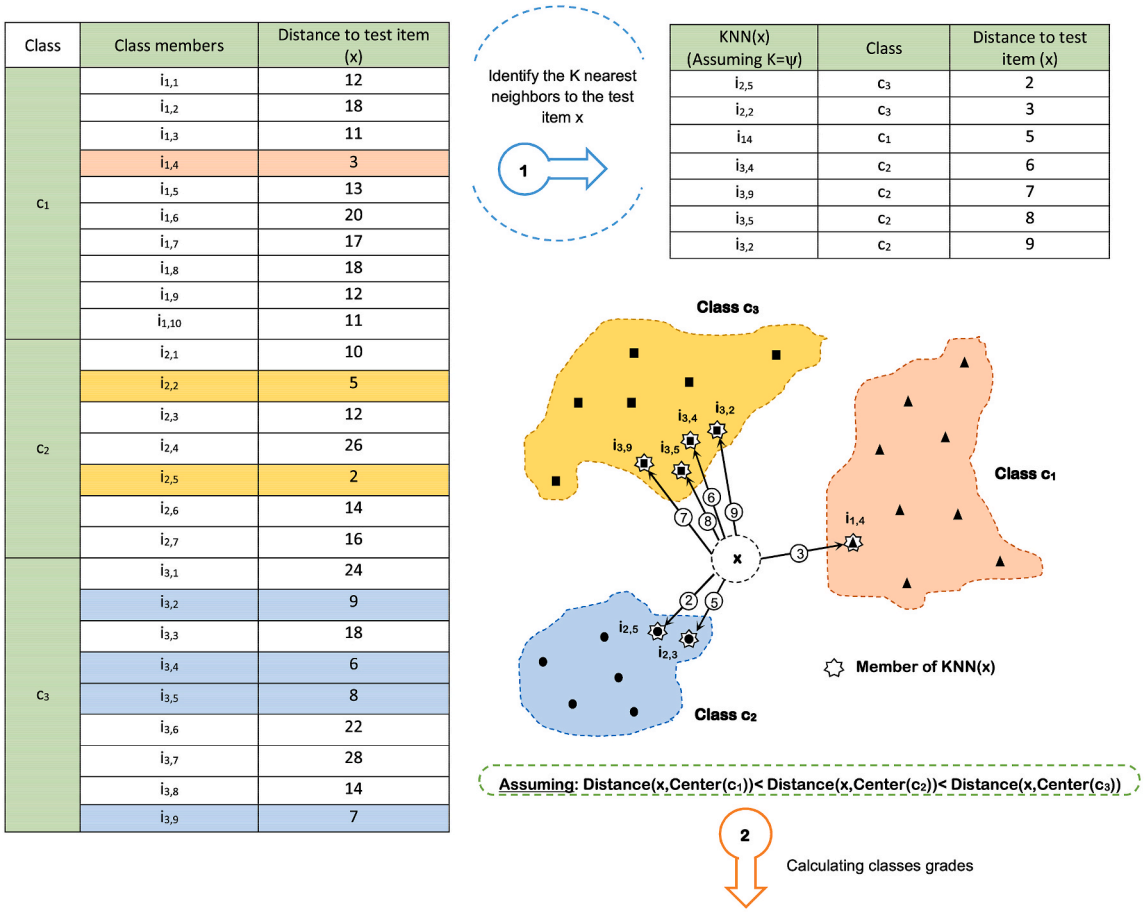
Fig. 11. Calculating Probability Based on Distance to Nearest Neighbors (illustrative example).

once. On the other hand, additional time penalty is added in the case of AKNN as it repeats the calculations accumulatively. Accordingly, the basic problem in AKNN is that it consumes more time to detect the target class of the input test item. However, in the applications that the time is not critical compared to the accuracy such as Covid-19 diagnose, this problem has no importance since the diagnose accuracy has the most priority.

5.3.4. Calculating the proper values of αα, ββ, and λ

The critical challenge now is how to set the proper values of the tuning parameters αα, ββ, and λ to guarantee the maximum classification accuracy. To solve this issue, consider the equation $Q = q_1Aa + q_2Ba + q_3Ca$, which is a linear equation in three variables. The variables are *Aa, Ba,* and *Ca*, while q_1, q_2, and q_3 are the equation coefficients (weights). The number Q is the constant of the equation. The solution of this equation is a specific point in $\mathbb{R}^3$ such that when the *Aa* coordinate of the point is multiplied by q_1, the *Ba* coordinate of the point is multiplied by q_2, the *Ca* coordinate of the point is multiplied by q_3, and then those three products are added together, the answer equals Q. However, usually, there are infinite solutions of a linear equation of three variables. The set of solutions in $\mathbb{R}^3$ of a linear equation of three variables is a two dimensional plane as illustrated in Fig. 13. The weights q_1, q_2, and q_3 determine the orientation of the solution plane. The problem now is to set the suitable values of q_1, q_2, and q_3 to get the suitable orientation of the solution plane.

As depicted in the linear equation in three variables illustrated in Fig. 13, q_1, q_2, and q_3 are the weights in *Aa, Ba,* and *Ca* dimensions respectively. Hence, if the value of a specific dimension is effective, the corresponding weight should be increased. For illustration, if the value in the *Aa* dimension is more important (effective) than the values in *Ba*



k	KNN(x)			Class Share (CS)			Target_Class(x)	Class Grade		
	Item Added	Class	Distance to x	c_1	c_2	c_3		c_1	c_2	c_3
Initially (k=0)	None	None	None	0	0	0	None	0	0	0
k=1	$i_{2,5}$	c_2	2	0	1	0	c_2	0	1	0
k=2	$i_{1,4}$	c_1	3	1	1	0	c_1	1	1	0
k=3	$i_{2,3}$	c_2	5	1	2	0	c_2	1	2	0
k=4	$i_{3,4}$	c_3	6	1	2	1	c_2	1	3	0
k=5	$i_{3,9}$	c_3	7	1	2	2	c_2	1	4	0
k=6	$i_{3,5}$	c_3	8	1	2	3	c_3	1	4	1
k=7	$i_{3,2}$	c_3	9	1	2	4	c_3	1	4	2

Fig. 12. Calculating Probability Based on K Nearest Neighbors (illustrative example).

and *Ca* dimensions, the weight q_1 should be greater than q_2 and q_3. Hence; $q_1 \propto effectiveness(Aa)$, $q_2 \propto effectiveness(Ba)$, and $q_3 \propto effectiveness(Ca)$. Now, consider (24); $Belonging(x|c_i) = \xi * P_{DTCC}(x|c_i) + \beta\beta * P_{DTNN}(x|c_i) + \lambda * P_{AKNN}(x|c_i)$, it can be concluded that;

$$\xi \propto effectiveness(P_{DTCC}(x|c_i)) \Rightarrow \xi \approx effectiveness(P_{DTCC}(x|c_i))$$

$$\beta\beta \propto effectiveness(P_{DTNN}(x|c_i)) \Rightarrow \beta\beta \approx effectiveness(P_{DTNN}(x|c_i))$$

$$\lambda \propto effectiveness(P_{AKNN}(x|c_i)) \Rightarrow \lambda \approx effectiveness(P_{AKNN}(x|c_i))$$

We assume that $effectiveness(P_T(x|c_i))$ as the classification accuracy if $P_T(x|c_i)$ is the only considered probability for classifying the test item x as; $Target(x) = \underset{\forall c_i \in C}{argmax}(P_T(x|c_i))$ where T refers to DTCC, DTNN, or AKNN. Hence, the proper values of ξ, ββ, and λ can be calculated empirically by using a set of labeled items, the classification accuracy can be calculated, then the proper values of ξ, ββ, and λ can be identified accordingly.

6. Experimental results

In this section, the effectiveness of Distance Based Classification Strategy (DBCS) strategy is examined. CIHI is aided by a novel classification strategy called Distance Based Classification Strategy (DBCS), which uses the Covid-19 dataset to identify people who are vulnerable to Covid-19 infection. Individuals are classified into 6 types using the proposed DBCS, and appropriate preventative actions can be performed for each type. DBCS is comprised of three successive phases; ORP, FSP, and CP. The HOR technique will be given in ORP in order to rapidly and precisely reject outliers by using standard division and IBPSO. FSP uses the HFS technique which includes Chi-square as a filter method and IBGWO as a wrapper method, to choose the most significant subset of features. Finally, CP uses Accumulative K-Nearest Neighbors (AKNN). Our implementation uses the "NileDS" dataset that consists of Covid-19 dataset [36]. In this work, 10-fold cross-validation method is used to analyze the classification performance. Based on the 10-fold cross-validation method, the NileDS dataset has been split into ten equal subsets of data. Nine of these ten subsets of data are used for training while the last one is used for testing. Hence, each case (patient)

appears nine times in a training set, and appears one time in a test set. Accordingly, the10-fold cross-validation method does not focus on how the cases are partitioned. Confusion matrix matrices will be used to evaluate DBCS [9,37]. The chosen value of K is assigned empirically. A basic classifier (KNN) is employed using different values of K using 1000 different cases of the employed dataset in which 800 cases is used for training and 200 cases for testing. For each value of K, the corresponding accuracy and error are calculated for the classifier. The optimal value of K is the one which maximizes the classification accuracy and accordingly minimizes the error rate. In our case, the used range is $K \in [1,40]$. As illustrated in Fig. 14, the best value of K, which minimizes the classification error rate is $K = 13$, hence, it is the used value through the next experiments.

6.1. NileDS dataset description

Finding a dataset to be used in the study of Covid-19 is very difficult, because this disease is a relatively new form of coronavirus. We devise a Web-based form to collect routine blood tests from people before and after their infection with Covid-19 in order to overcome this obstacle. This form is associated to the Nile Higher Institute's artificial intelligence lab and can be found at [36]. As shown in Table 7 (a-c), the NileDS dataset contains 50 features collected from numerical laboratory tests. After applying the FS and AS stages in feature selection stage, the number of most important features became 34. The overall number of people who have filled out the form until now is 2215, with 1389 infected people, 396 unconfirmed cases, and 430 un-covid-19 people as shown in Table 8. The available dataset is divided into two classes; training data (70% of all data) and testing data (30% of all data). Patients in the dataset can be classified into six types (Type (A-F)) based on their individual Covid-19 vulnerability levels. A screenshot from the NileDS dataset is shown in Fig. 15.

6.2. Evaluation performance metrics

In the NileDS dataset, the Covid-19 patients are classified into six classes based on the expected resistance of their body against the Corona virus. If a patient was infected with it, these classes indicate how much damage the virus may cause if the person is actually infected, while other patients were considered as negative. The performance of the classification strategy will be tested using five metrics in the following experiments. The metrics are accuracy, error, recall, precision, and run-time. The confusion matrix can be used to calculate the values of these parameters [9,37]. As shown in Table 9, a confusion matrix and associated formulas were used.

6.3. Testing the Distance Based Classification Strategy (DBCS)

This section will evaluate the DBCS, which is made up of three successive phases; (i) ORP using HOR technique, (ii) FSP using HFS method, and (iii) CP using AKNN method. After removing outliers in ORP using HOR approach, selecting the most important features in FSP using HFS approach, the filtered data is given to CP to correctly learn the classification model. Table 10 shows the performance of DBCS in terms of accuracy, error, precision, and recall based on 10-fold cross-validation. To demonstrate the efficiency of DBCD, it was compared with several previous Covid-19 classification methods including DBNB [4], FCNB [1], CNN [19], HDS [2], and CPDS [3]. Fig. (16 - 20) and Table 11 illustrate the five metrics; accuracy, error, precision, recall, and run-time of the employed strategies. The results show that DBCS outperforms other classification strategies in terms of accuracy, precision, recall, and run-time performance parameters.

According to Table 10, the performance of DBCS strategy in terms of accuracy, error, precision, and recall at each fold of 10-fold cross-validation is presented to classify people based on their vulnerability by Covid-19. Additionally, 10-fold values are averaged values across the four metrics. In Table 10, the best performance values for DBCS strategy are presented at 4,8,9, and10-fold while the worst values are presented at 1,2,3,5,6, and 7-fold. Additionally, the average values in terms of accuracy, error, precision, and recall are 0.90, 0.09, 0.85, and 0.81 respectively. Figures (16 → 19) and Table 11 show that at the number of training samples equal to 973 patients, the accuracy values provided by DBNB, FCNB, CNN, HDS, CPDS, and DBCS are 0.75, 0.71, 0.63, 0.64, 0.66, and 0.91 respectively. According to these results, DBCS obtains the highest accuracy value when two phases, named; the outlier rejection phase and the feature selection phase are used before implementing the classification model to classify Covid-19 patients based on the individual's level of vulnerability to Covid-19. The error values introduced by DBNB, FCNB, CNN, HDS, CPDS, and DBCS techniques are 0.25, 0.29, 0.36, 0.26, 0.34, and 0.09 respectively. Hence, DBCS can obtain the maximum accuracy value and the minimum error value. DBCS introduces precision value reaches to 0.86 while DBNB, FCNB, CNN, HDS, and CPDS provide precision values equal to 0.73, 0.59, 0.61, 0.63, and 0.55 respectively. The recall values of DBNB, FCNB, CNN, HDS, CPDS, and DBCS are 0.67, 0.61, 0.58, 0.63, 0.62, and 0.83 respectively. Fig. 20 illustrates that DBCS is faster than other strategies; DBNB, FCNB, CNN, HDS, and CPDS. Hence, DBCS has the highest accuracy and the fastest run time. Finally, figures (16 → 20) and Table 11 show that DBCS outperforms other previous strategies by providing the highest accuracy value, the lowest error value, and the fastest run time.

7. DBCS pros and cons

Several pros and cons of the proposed DBCS will be discussed in this section as presented in Table 12. According to Table 12, the proposed DBCS has many benefits such as it is a novel, efficient, applicable, scalable, accurate, and fast strategy. In fact, this paper is the first to provide the issue of predicting how people's bodies will react if they are infected with Covid-19. DBCS is a new prediction strategy used to classify people based on their vulnerability by Covid-19. Additionally, DBCS has high prediction efficiency because it employs three new methods called HOR to reject outliers, HFS to select the best subset of features, and AKNN to accurately classify people. DBCS has also the ability to be applied in hospitals and medical centers because of its simplicity and straightforward during the implementation. DBCS is scalable because it can handle datasets incrementally and it can be used to solve other prediction problems in the medical systems. DBCS can provide more accurate and fast predictions compared to other recent strategies. On the other hand, the proposed DBCS suffers from higher time delay compared with other competitors. The cause of such delay is concentrated in the use of the pre-processing phases, which are; outlier rejection and feature selection. However, this delay does not have an effect on the system performance because the implementation of both outlier rejection and feature selection methods is performed offline. Additionally, providing accurate diagnose is more important than getting fast diagnose.

8. Conclusions

A new strategy called Distance Based Classification Strategy (DBCS) has been provided in this paper for classifying individuals into six different types, then suitable precautionary measures can be taken for every type. DBCS composes of three phases called Outlier Rejection Phase (ORP), Feature Selection Phase (FSP), and Classification Phase (CP) to quickly give more accurate classifications. After eliminating outliers in ORP and selecting the most significant features in FSP, filtered data was passed to CP to learn the Accumulative K-Nearest Neighbors (AKNN) classification model. Hybrid Outlier Rejection (HOR) method that combines standard division and Improved Binary Particle Swarm Optimization (IBPSO) methods has been used in ORP to determine outliers and then reject them. On the other hand, Hybrid Feature Selection (HFS) method that consists of Chi-square as a filter method and

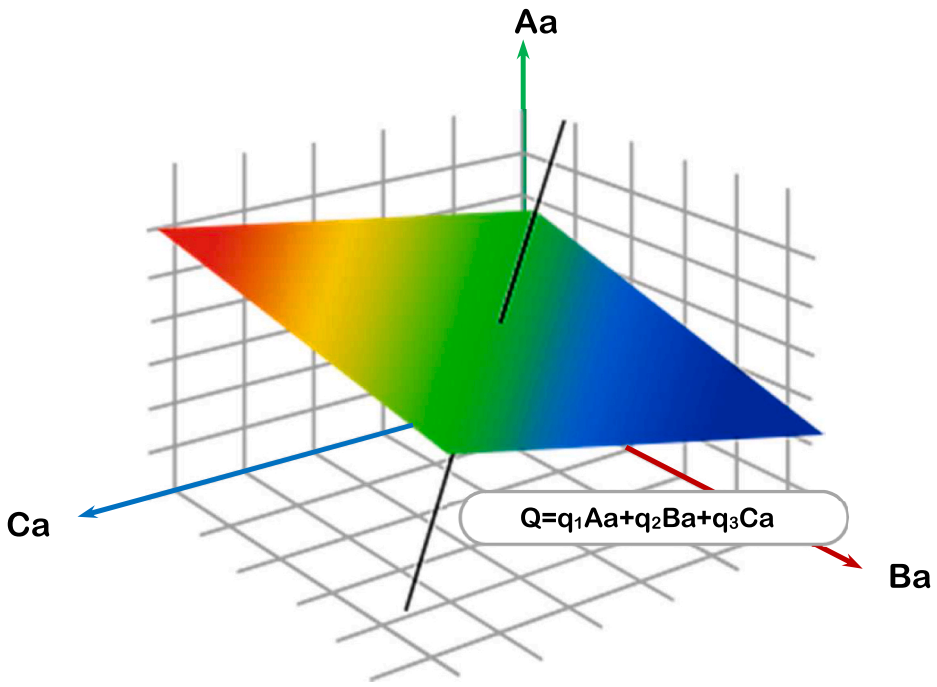


Fig. 13. Solution plan for the 3 variable equation.

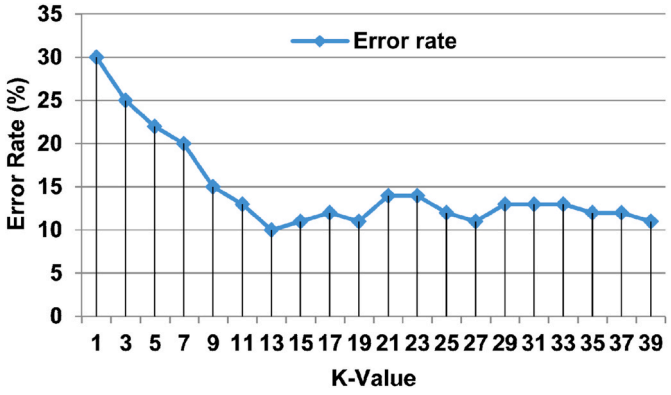


Fig. 14. Error rate VS K-value.

Improved Binary Gray Wolf Optimization (IBGWO) as a wrapper method has been used in FSP to select the most informative features for improving the performance of classification model and avoiding overfitting. In CP, the core of such classification strategy is a proposed AKNN classifier that was used to quickly and accurately classify patients based on the forward data from ORP and FSP. Experimental results showed that the proposed DBCS provided fast and accurate results compared to the existing methods in terms of accuracy, error, precision, and recall. DBCS introduced accuracy, error, precision, and recall values reach to 0.91, 0.09, 0.86, and 0.84 respectively. Hence, the proposed DBCS provided the best accuracy value that is higher than other recent strategies. Finally, the proposed DBCS based on ORP, FSP, and AKNN in CP introduced fast and more accurate results than the existing techniques in terms of accuracy, precision, recall, and execution time.

9. Future works

In the future, there are many different directions through which the efficiency of the classification strategy proposed in the research can be improved such as; (i) implementing more heuristics such as the deep learning and fuzzy inference methods as well as different classifiers such as; support vector machines in the proposed classification method to obtain more accurate classifications, (ii) testing the efficiency the proposed DBCS using different datasets collected from different regions and at different sizes. Thus, it is possible to accurately determine the efficiency of the proposed strategy, as it is known that the natures and characteristics of human bodies differ from each other at different regions, and therefore the ability of people to resist diseases may differ according to the genetic characteristics and the nature of the environment in which they live, (iii) implementing the presented strategy on

Table (7-a)
Descriptions about the features of NileDS dataset.

Feature	Description	Selected Feature
Age	**Age of the patient.**	**Yes**
Gender	**Male/Female**	**No**
Glucose	**Glucose is the most common form of sugar detected in the blood.**	**Yes**
Blood type	**Find out what type of blood you have.**	**Yes**
Blood Pressure	**It is the pressure exerted on the artery walls.**	**Yes**
Body Mass Index (BMI)	**BMI is a metric that reflects how much fat is in your body. It's used to determine if someone is at a healthy weight.**	**Yes**
Diabetes Pedigree Function	**A function that assesses the risk of diabetes depending on a person's family history.**	**No**
Total_Bilirubin	**It is a test that examines the quantity of bilirubin in the blood and is used to assess liver function.**	**Yes**
Direct_Bilirubin	**It is a test that searches for bilirubin in the urine or blood to determine the amount of conjugated bilirubin.**	**Yes**
Alkaline_ Phosphotase	**It's a measurement of how much alkaline phosphotase is in your blood. Alkaline_ Phosphotase is an enzyme that's located all over the body. It can be found primarily in the liver, digestive system, kidneys, and bones.**	**Yes**
Alamine_ Aminotransferase (ALT)	**ALT is a liver and kidney enzyme that is normally found in the cells. When ALT levels in the blood are high, it indicates that the liver has been damaged.**	**Yes**
Aspartate_ Aminotransferase (AST)	**The enzyme AST is generally found in the liver and the heart. When AST levels in the blood are increased, it can signify liver illness, cardiac disease, or pancreatitis.**	**Yes**
Total_ Protiens	**It's a measurement of how much protein is in your blood. When total protein levels are high, it could be a sign of dehydration or malignancy.**	**No**
Albumin	**It is a protein produced by the liver, this test evaluates the amount of albumin in the blood**	**Yes**
Globulin_Ratio	**It is the percentage of albumin to globulin in blood plasma.**	**Yes**
Red blood count	**It's a blood test that determines how many red blood cells have haemoglobin that transports oxygen throughout the body.**	**Yes**
Pus Cell count	**It is a type of white blood cell (neutrophil) present in pus.**	**No**
Bacteria test	**Bacteria test are used to aid in the diagnosis of infections whose organisms are not apparent to the human eye**	**Yes**
Blood urea test	**It is used to evaluate the concentration of nitrogen in the blood. When your blood urea level rises, it indicates your kidneys are unable to properly eliminate urea from the bloodstream.**	**Yes**

Table (7-b)
Descriptions about the features of NileDS dataset.

Feature	Description	Selected Feature
Serum creatinine	**Measure of muscle metabolism that indicates kidney health.**	**Yes**
Sodium	**A sodium test used to determine the level of sodium in the blood during a blood test.**	**No**
Potassium	**It is a part of blood test that is used to determine how much potassium is in the blood.**	**Yes**
Haemoglobin	**It is a blood test that is used to determine the quantity of haemoglobin in our blood.**	**Yes**
Packed cell volume	**It is a test that is used to determine if a patient has dehydration, anaemia, or polycythaemia.**	**No**
White blood cell count	**White blood cells make up the immune system where they help fight infections and other diseases.**	**Yes**
Hypertension	**It is a condition in which the blood arteries have consistently high pressure. Hypertension is a serious medical condition that can endanger kidneys, heart, brain, and other organs.**	**Yes**
Pedal Edema	**Edema is the medical term for swelling. Body parts swell as a result of injuries and inflammation.**	**No**
Resting electrocardiographic results	**It is a medical test that predicts the heart disease by measuring electrical activity of the heart.**	**Yes**
Anemia	**Anemia is a medical condition in which the quantity of haemoglobin or red blood cells is low.**	**No**
Diabetes mellitus	**Diabetes mellitus is a metabolic disorder that impairs the body's ability to produce sugar.**	**Yes**
Coronary artery disease	**The coronary arteries transport oxygen, nutrients, and blood to your heart. Coronary artery disease happens when the principal blood arteries in your heart become damaged.**	**Yes**
Appetite	**The appetite test is used to determine a person's appetite.**	**No**
Maximum heart rate achieved	**The heart rate is determined by the number of beats of the heart per minute bpm).)**	**Yes**
Exercise induced angina	**Angina is a type of chest pain caused by exercise, stress, or other causes that force the heart to work harder. It's a pretty common symptom of coronary artery disease.**	**Yes**
Atherosclerosis	**Arteriosclerosis happens when arteries that supplies blood from your heart to the body thicken and stiffen, resulting in decreased blood flow to your tissues and organs.**	**Yes**
D-Dimer test	**A protein fragment known as a D-dimer is generated when a blood clot melts in your body. The test shows if D-dimer is presents in the blood or not.**	**Yes**
C-Reactive Protein (CRP)	**The liver produced the CRP and the test is used to monitor or identify the inflammatory illnesses.**	**Yes**
Lactate Dehydrogenase (LDH)	**The LDH test is used to diagnose tissue injury.**	**Yes**
Troponin	**Troponins are a protein family that regulates muscle contraction in skeletal and cardiac muscle fibres. Troponin assays measure the amount of cardiac-specific troponin in the blood to diagnose heart injury.**	**Yes**

Table (7-c)
Descriptions about the features of NileDS dataset.

Feature	Description	Selected Feature
Platelets Count (PC)	**The platelet count (PC) is a blood test that determines the average number of platelets in a person's blood. Platelets help to heal wounds and avoid dangerous bleeding in the bloodstream.**	**Yes**
Neutrophils Count (NC)	**Neutrophils are a type of WBC that form about (50–75%) of the total. NC provides vital information on the patient's health condition.**	**Yes**
Lymphocytes Count (LC)	**The LC test determines the lymphocyte count, which is a component of WBC.**	**Yes**
Monocytes count	**The monocytes count test determines the number of monocytes circulating in the human blood.**	**No**
Eosinophil	**Eosinophil is a part of the immune system that aids in disease prevention by avoiding infections and boosting inflammation.**	**No**
Basophils	**Basophils are formed from bone marrow that and help the immune system work properly.**	**No**
Gamma-Glutamyl Transpeptidas (GGT)	**GGT is an enzyme that is found all over the body. GGT levels in the blood can be used to diagnose bile duct damage or liver illness.**	**No**
Chest pain type	**The presence of abnormal pain between the base of the neck and diaphragm is described as chest pain.**	**No**
Fasting blood sugar	**This test determines the amount of sugar in a blood after a fasting state.**	**No**
Ferritin	**Ferritin is responsible for iron storage, and the high value indicates the presence of severe inflammation.**	**No**
Creatine phosphokinase (CPK)	**CPK is a protein present in the skeletal muscles, brain, and heart. CPK aids in the induction of chemical changes that happen in human body.**	**Yes**

Table 8
Distribution of cases in the NileDS dataset based on their types.

Criteria	Value/Description		
Total number of cases	Covid-19 Patients	Un-Covid-19 People	Un-confirmed cases
	1389	**430**	**396**
Type of Covid-19 Patients	Type A	Type B	Type C
	173	**239**	**129**
	Type D	Type E	Type F
	228	**471**	**149**

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Blood type	Age	Gender	Glucose	BloodPressure	BMI	.....	Alkaline_ Phosphotase	Alamine_ Aminotransferase	Aspartate_ Aminotransferase	.....	....	white blood cell count	red blood cell count	hypertension	coronary artery disease	classification	Type
2	O+	50	Male	148	72	33.6	.....	202	22	19	.....	....	7900	3.9	yes	no	UN-confirmed	
3	A+	31	Male	85	66	26.6	.....	210	51	59	.....	....			no	no	COVID	B
4	AB-	32	Male	183	64	23.3	.....	260	31	56	.....	....			yes	no	COVID	D
5	A+	21	Male	89	66	28.1	.....	183	91	53	.....	....	7200	5.5	no	no	un-covid	
6	B-	33	Male	137	40	43.1	.....	342	168	441	.....	....	8300	4.6	yes	no	UN-confirmed	
7	O+	30	Female	116	74	25.6	.....	165	15	23	.....	....	4200	3.4	yes	no	COVID	A
8	O-	26	Female	78	50	31	.....	293	232	245	.....	....	9900	4.7	no	no	COVID	A
9	AB-	34	Male	168	74	38	.....	269	58	45	.....	....	10500	6.1	no	no	COVID	B
10	B-	57	Male	139	80	27.1	.....	298	33	59	.....	....	2200		no	no	COVID	F
11	A+	59	Male	189	60	30.1	.....	161	27	28	.....	....	7200	2.6	yes	yes	un-covid	
12	A-	51	Male	166	72	25.8	.....	243	21	23	.....	....	7500	5.6	no	no	COVID	D
13	AB+	32	Male	100	0	30	.....	486	25	34	.....	....	4200	3.3	yes	yes	UN-confirmed	
14	AB+	27	Female	126	88	39.3	.....	187	16	18	.....	....	8400	5.5	no	no	COVID	E
15	B-	50	Male	99	84	35.4	.....	699	64	100	.....	....			no	no	UN-confirmed	
16	AB+	33	Male	103	30	43.3	.....	490	60	68	.....	....			no	no	un-covid	
17	O+	32	Male	115	70	34.6	.....	320	28	56	.....	....			yes	no	COVID	A
18	A-	26	Male	180	64	34	.....	237	18	28	.....	....	15700	3.8	no	no	COVID	A
19	AB-	37	Female	133	84	40.2	.....	199	34	31	.....	....	10500	5	no	no	COVID	A
20	B-	48	Female	106	92	22.7	.....	201	18	22	.....	....	7900	4.5	no	no	COVID	E

Fig. 15. A snapshot from the NileDS dataset.

Table 9
Confusion matrix and its formulas.

		Predicted Class		
		True	False	
Actual Class	**True**	**True Positive cases (TPc)**	**False Negative cases (FNc) Type 2 Error**	Accuracy **= (TPc + TNc)/ (TPc + TNc + FPc + FNc)**
	False	**False Positive cases (FPc) Type 1 Error**	**True Negative cases (TNc)**	Error **= 1-Accuracy**
Recall **= TPc/ (TPc + FNc)**	Precision **= TPc/(TPc + FPc)**			

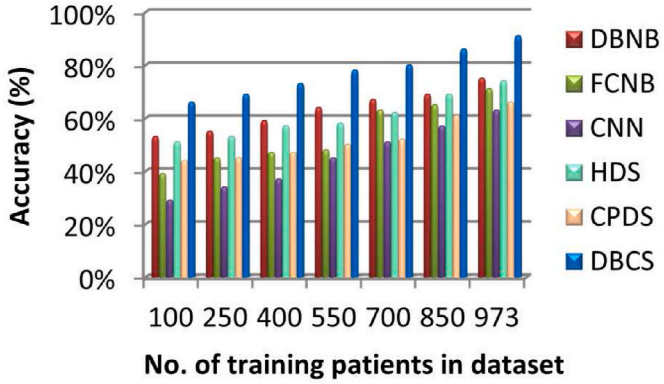


Fig. 16. Accuracy of several strategies of Covid-19 detection strategies.

Table 10
The performance of DBCS in terms of accuracy, error, precision, and recall based on 10-fold cross-validation.

Fold	Accuracy	Error	Precision	Recall
1	90.25%	9.75%	84.25%	81.27%
2	90.50%	9.50%	85.50%	81.50%
3	89%	11.00%	82%	82%
4	91%	9.00%	84%	81%
5	89.98%	10.02%	85.98%	78.98%
6	90%	10.00%	85.98%	75.98%
7	90.02%	9.98%	84.99%	79.99%
8	91%	9.00%	86%	82.25%
9	91%	9.00%	83%	83%
10	91%	9.00%	86%	83%
Average	90.38%	9.62%	84.77%	80.90%

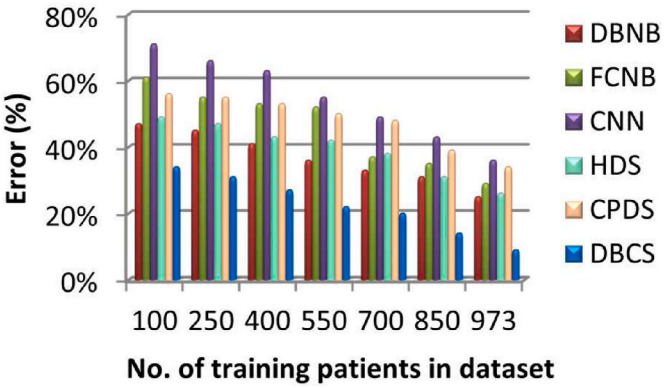


Fig. 17. Error of several strategies of Covid-19 detection.

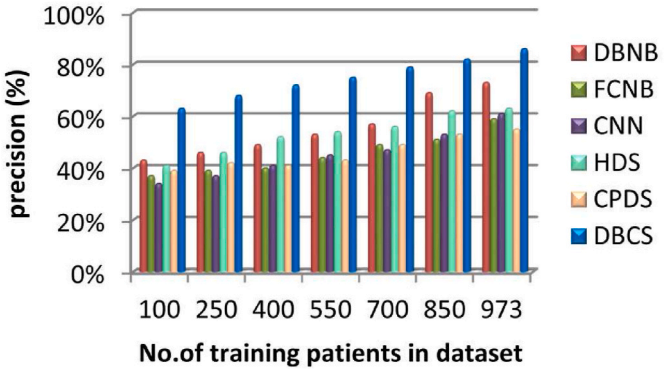


Fig. 18. Precision of several strategies of Covid-19 detection.

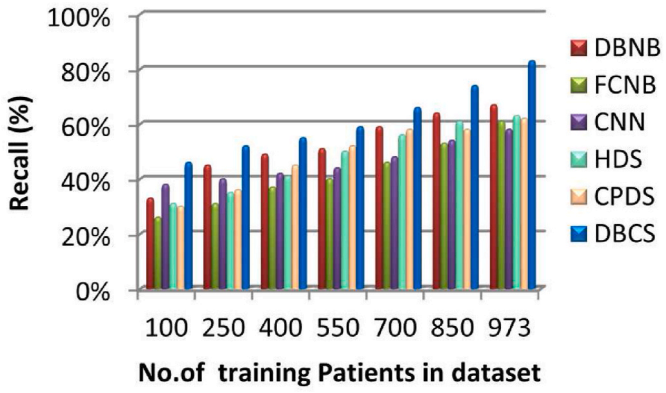


Fig. 19. Recall of several strategies of Covid-19 detection.

Table 11
Comparison between DBCS and other recent classification strategies in terms of accuracy, precision, and recall.

Used technique	Accuracy	Error	Precision	Recall
DBNB	75%	25%	73%	67%
FCNB	71%	29%	59%	61%
CNN	63%	36%	61%	58%
HDS	64%	26%	63%	63%
CPDS	66%	34%	55%	62%
DBCS	91%	9%	86%	83%

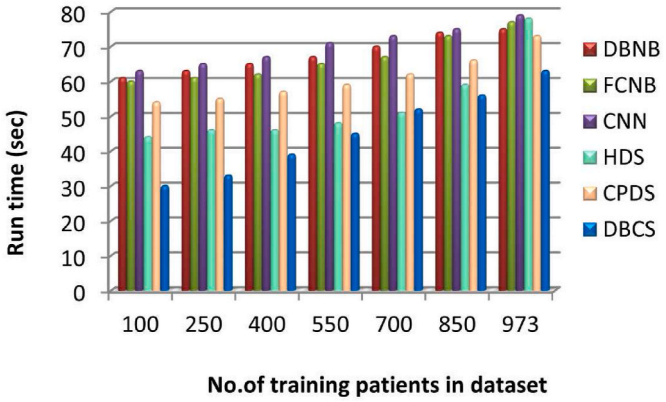


Fig. 20. Run time of several strategies of Covid-19 detection.

Table 12
Pros and Cons of the proposed DBCS.

DBCS Pros		DBCS Cons	
Property	Description	Property	Description
Novelty	• DBCS is the first to handle the issue of predicting how people's bodies will react if they are infected with Covid-19. Hence, the research idea is novel. • The proposed DBCS itself is a new prediction strategy, which can be used to accurately classify people based on their vulnerability by Covid-19.	**Delay**	• The proposed DBCS has a high delay because it contains outlier rejection and feature selection phases. In fact, the effect of this delay is ignored because both phases are performed offline and this paper aims to provide accurate results more than fast results.
Accuracy	• DBCS has high prediction accuracy compared to other recent strategies because it contains three new methods that work smoothly with each other, which are HOR, HFS, and AKNN methods. Accordingly, the whole strategy (e.g., DBCS) provides a high efficiency.		
Applicability	• DBCS has the ability to be applied in hospitals and medical centers because it is simple and straightforward and easy to be implemented.		
Scalability	• DBCS is scalable because it can handle data set incrementally. • It can be used to solve other prediction problems in the medical systems.		

hardware device (Application Specific Integrated Circuits) such as; Field Programmable Gate Array (FPGA) and Digital Signal Processor (DSP). Hence, we can produce a standalone device for diagnosing and classifying people based on how much their body will react if they are infected with Covid-19. This will certainly be useful in containing the Corona pandemic.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to express their sincere appreciation to Mansoura University for the generous support, specially to the President of Mansoura University for his unending support and gentleness. Also, a special thanks to Dr. Salah Abd Elghafar Mansour for his unending support and generosity. We would also like to express our heartfelt appreciation and gratitude to Dr. Omnia Anees Mansour for assisting us throughout this research and providing us with all of the medical information that we require.

References

[1] N. Mansour, A. Saleh, M. Badawy, H. Ali, Accurate detection of covid-19 patients based on feature correlated Naïve Bayes (FCNB) classification strategy, in: Journal of Ambient Intelligence and Humanized Computing, Springer, 2021, pp. 1–33. https://link.springer.com/article/10.1007/s12652-020-02883-2.

[2] W. Shaban, A. Rabie, A. Saleh, M. Abo-Elsoud, Detecting covid-19 patients based on fuzzy inference engine and deep neural network, in: Applied Soft Computing, 99, Elsevier, 2021, pp. 1–19.

[3] W. Shaban, A. Rabie, A. Saleh, M. Abo-Elsoud, A New COVID-19 patients detection strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier, in: Knowledge-Based Systems, 205, Elsevier, 2020, pp. 1–18.

[4] W. Shaban, A. Rabie, A. Saleh, M. Abo-Elsoud, Accurate Detection Of COVID-19 Patients Based On Distance Biased Naıve Bayes (DBNB) Classification Strategy, in: Pattern Recognition, Elsevier, 2021, https://doi.org/10.1016/j.patcog.2021.108110.

[5] P. Zhai, Y. Ding, X. Wu, J. Long, et al., The epidemiology, diagnosis and treatment of COVID-19, in: International Journal of Antimicrobial agents, 55, Elsevier, 2020, pp. 1–13. Issue 5.

[6] G. Kim, M. Kim, S. Ra, J. Lee, et al., Clinical characteristics of asymptomatic and symptomatic patients with mild COVID-19, in: Clinical Microbiology and Infection, 26, Elsevier, 2020, p. 948. Issue 7.

[7] Z. Wu, J. McGoogan, Characteristics of and important lessons from the coronavirus disease 2019 (COVID-19) outbreak in China, J. Am. Med. Assoc. 323 (Issue13) (2020) 1239–1242.

[8] A. Rabie, A. Saleh, K. Abo-Al-Ez, A new strategy of load forecasting technique for smart grids, Int. J. Mod. Trends Eng. Res. (IJMTER) 2 (Issue 12) (2015) 332–341.

[9] A. Saleh, A. Rabie, K. Abo-Al-Ezb, A data mining based load forecasting strategy for smart electrical grids, in: Advanced Engineering Informatics, 30, Elsevier, 2016, pp. 422–448. Issue 3.

[10] A. Rabie, S. Ali, H. Ali, A. Saleh, A fog based load forecasting strategy for smart grids using big electrical data, in: Cluster Computing, 22, Springer, 2019, pp. 241–270. Issue 1.

[11] A. Rabie, S. Ali, A. Saleh, H. Ali, A new outlier rejection methodology for supporting load forecasting in smart grids based on big data, in: Cluster Computing, 23, Springer, 2020, pp. 509–535.

[12] A. Rabie, S. Ali, A. Saleh, H. Ali, A fog based load forecasting strategy based on multi-ensemble classification for smart grids, in: Journal of Ambient Intelligence and Humanized Computing 11, Springer, 2020, pp. 209–236. Issue 1.

[13] A. Rabie, A. Saleh, H. Ali, Smart Electrical Grids Based on Cloud, IoT, and Big Data Technologies: State of the Art, in: Journal of Ambient Intelligence and Humanized Computing, Springer, 2020, pp. 1–32, https://doi.org/10.1007/s12652-020-02685-6.

[14] Y. Fang, H. Zhang, J. Xie, et al., Sensitivity of chest CT for COVID-19: comparison to RT-PCR, Radiology (2020), https://doi.org/10.1148/radiol.2020200432.

[15] M. Barstugan, U. Ozkaya, and S. Ozturk, "Coronavirus (COVID-19) Classification Using CT Images by Machine Learning Methods," arXiv preprint arXiv:2003.09424.

[16] Zhonghua Liu Xing Bing Xue Za Zhi, The epidemiological characteristics of an outbreak of 2019 novel coronavirus diseases (COVID-19) in China, Novel Coronavirus Pneumonia Emerg. Response Epidemiol. Team 41 (Issue 2) (2020) 145–151, https://doi.org/10.3760/cma.j.issn.0254-6450.2020.02.003.

[17] S. Hoehl, H. Rabenau, A. Berger, M. Kortenbusch, et al., Evidence of SARS-CoV-2 infection in returning travelers from Wuhan, China, N. Engl. J. Med. 41 (Issue 2) (2020), https://doi.org/10.1056/NEJMc2001899.

[18] C. Huang, Y. Wang, X. Li, L. Ren, et al., Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China, Lancet 395 (10223) (2020) 497–506, https://doi.org/10.1016/S0140-6736(20)30183-5.

[19] H. Maghdid, A. Asaad, K. Ghafoor, A. Sadiq, et al., Diagnosing COVID-19 Pneumonia from X-Ray and CT images using deep learning and transfer learning algorithms, 2020, pp. 1–8, arXiv preprint arXiv:2004.00038.

[20] Q. Li, J. Ning, J. Yuan, L. Xiao, A depthwise separable dense convolutional network with convolution block attention module for COVID-19 diagnosis on CT scans, in: Computers in Biology and Medicine 137, Elsevier, 2021, pp. 1–13.

[21] S. Serte, H. Demirel, Deep learning for diagnosis of COVID-19 using 3D CT scans, in: Computers in Biology and Medicine 132, Elsevier, 2021, pp. 1–8.

[22] V. Tellis, D. Souza, Detecting Anomalies In Data Stream Using Efficient Techniques : A Review, in: Proceedings of the 2018 International Conference on Control, Power, Communication and Computing Technologies, ICCPCCT, Kannur, India, 2018, pp. 296–298.

[23] C. Park, Outlier and anomaly pattern detection on data streams, in: The Journal of Supercomputing, Springer, 2018, pp. 1–11, https://doi.org/10.1007/s11227-018-2674-1.

[24] Z. Shou, S. Li, Large dataset summarization with automatic parameter optimization and parallel processing for local outlier detection, in: Concurrency Computation Practice and Experience, 30, Wiley, 2018, pp. 1–13. Issue 23.

[25] J. Posio, K. Leiviskä, J. Ruuska, P. Ruha, Outlier detection for 2D temperature data, in: IFAC proceedings volumes, 41, Elsevier, 2008, pp. 1958–1963. Issue 2.

[26] A. Karale, M. Lazarova, P. Koleva, V. Poulkov, MEOD: memory-efficient outlier detection on streaming data, Symmetry 458 (13) (2021) 1–11, https://doi.org/10.3390/sym13030458.

[27] A. Mohemmed, M. Zhang, W. Browne, Particle swarm optimisation for outlier detection, in: " Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, 2010, pp. 83–84, https://doi.org/10.1145/1830483.1830498.

[28] Y. Zhang, N. Meratnia, P. Havinga, Outlier detection techniques for wireless sensor networks: a survey, IEEE Commun. Surv. Tutorials 12 (Issue 2) (2010) 159–170.

[29] N. Yu, L. Zhang, Y. Ren, A novel D-S based secure localization algorithm for wireless sensor networks, in: Security and Communication Networks, 7, Wiley, 2014, pp. 1945–1954. Issue 11.

[30] S. Gu, R. Cheng, Y. Jin, Feature selection for high-dimensional classification using a competitive swarm optimizer, in: Soft Computing, 22, Springer, 2018, pp. 811–822. Issue 3.

[31] S. Ayyad, A. Saleh, L. Labib, A new distributed feature selection technique for classifying gene expression data, Int. J. Biomath. (IJB) 12 (Issue 2) (2019) 1–34.

[32] S. Ayyad, A. Saleh, L. Labib, Gene expression cancer classification using modified K-Nearest Neighbors technique, in: BioSystems 176, Elsevier, 2019, pp. 41–51.

[33] M. Abdel-Basset, D. El-Shahat, I. El-henawy, V. de Albuquerque, S. Mirjalili, A new fusion of grey wolf optimizer algorithm with a two-phase mutation for feature selection, in: Expert Systems with Applications 139, Elsevier, 2020, pp. 1–14.

[34] S. Mirjalili, S. Mirjalili, A. Lewis, Grey wolf optimizer, in: Advances in engineering software, 69, Elsevier, 2014, pp. 46–61.

[35] E. El-kenawy, M. Eid, M. Saber, A. Ibrahim, MbGWO-SFS: Modified Binary Grey Wolf Optimizer Based on Stochastic Fractal Search for Feature Selection, in: IEEE Access, 8, IEEE, 2020, pp. 107635–107649.

[36] http://covid19.nilehi.edu.eg/Available_datasets.php.

[37] S. Visa, B. Ramsay, A. Ralescu, E. Knaap, Confusion matrix-based feature selection, in: Proceedings of the Twenty Second Midwest Artificial Intelligence and Cognitive Science Conference, MAICS, Cincinnati, OH, USA, 2011, pp. 120–127.